

**KERJA PRAKTIK – KI141330**

**Pengembangan Fitur Manajemen dan Evaluasi Karyawan  
pada Aplikasi *Learning Management System* Menggunakan  
Kerangka Kerja Laravel**

**PT. QAREER HARAPAN ASIA**

**Jl. Palmerah Barat No.48, Palmerah, Jakarta Barat, DKI  
Jakarta**

**Periode: 22 Juni 2020 - 22 Juli 2020**

**Oleh:**

Raden Bimo Rizki Prayogo (05111740000139)

Pembimbing Jurusan

Dr.tech.Ir. Raden Venantius Hari Ginardi, M.Sc.

Pembimbing Lapangan

Muhammad Rizki

DEPARTEMEN INFORMATIKA

Fakultas Teknologi Informasi dan Komunikasi

Institut Teknologi Sepuluh Nopember

Surabaya 2020

*[Halaman ini sengaja dikosongkan]*



**KERJA PRAKTIK – KI141330**

**Pengembangan Fitur Manajemen dan Evaluasi Karyawan  
pada Aplikasi Learning Management System Menggunakan  
Kerangka Kerja Laravel**

**PT. QAREER HARAPAN ASIA**

**Jl. Palmerah Barat No.48, Palmerah, Jakarta Barat, DKI  
Jakarta**

**Oleh:**

**Pembimbing Departemen**

Dr.tech.Ir. Raden Venantius Hari Ginardi, M.Sc

**Pembimbing Lapangan**

Muhammad Rizki

DEPARTEMEN INFORMATIKA

Fakultas Teknologi Informasi dan Komunikasi

Institut Teknologi Sepuluh Nopember

Surabaya 2020

*[Halaman ini sengaja dikosongkan]*

# LEMBAR PENGESAHAN

## KERJA PRAKTIK

### **Pengembangan Fitur Manajemen dan Evaluasi Karyawan pada Aplikasi *Learning Management System* Menggunakan Kerangka Kerja Laravel**

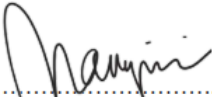
Oleh:

Raden Bimo Rizki Prayogo


05111740000139

Disetujui oleh Pembimbing Kerja Praktik:

1. Dr.tech.Ir. Raden Venantius  
Hari Ginardi, M.Sc  
NIP. 196505181992031003

  
.....  
(Pembimbing Jurusan)

2. Muhammad Rizki

  
.....  
(Pembimbing Lapangan)

SURABAYA, AGUSTUS, 2020

*[Halaman ini sengaja dikosongkan]*

**Pengembangan Fitur Manajemen dan Evaluasi Karyawan  
pada Aplikasi *Learning Management System*  
Menggunakan Kerangka Kerja Laravel**

**Nama Mahasiswa : Raden Bimo Rizki Prayogo**  
**NRP : 05111740000139**  
**Departemen : Informatika FTIK-ITS**  
**Pembimbing Jurusan : Dr.tech.Ir. Raden Venantius Hari  
Ginardi, M.Sc.**  
**Pembimbing Lapangan : Muhammad Rizki**

## ABSTRAK

*Pada saat ini, Qareer sedang mengembangkan suatu website Learning Management System (LMS) bernama kelas.center untuk memberikan layanan bagi berbagai perusahaan untuk melatih karyawannya secara daring.*

*Suatu LMS memerlukan berbagai fitur untuk melakukan evaluasi terhadap karyawan yang sedang dilatih dan melakukan manajemen terhadap kinerja karyawan yang sedang dilatih oleh pihak yang berwenang. Proyek Kerja Pratik ini bertujuan mengembangkan berbagai fitur tersebut untuk mempermudah penggunaan website tersebut. Manfaat yang dapat diperoleh dengan adanya pengembangan berbagai fitur manajemen karyawan pada aplikasi LMS kelas.center adalah melakukan evaluasi terhadap karyawan yang sedang dilatih. dan mempermudah manajemen karyawan oleh pihak yang berwenang. Fitur ini diimplementasikan dengan kerangka kerja Laravel dan menggunakan basis data MySQL.*

***Kata kunci: Learning Management System, Laravel, MySQL.***



## KATA PENGANTAR

Puji syukur kami haturkan kepada Allah SWT karena berkat rahmat-Nya kami dapat melaksanakan salah satu kewajiban kami sebagai mahasiswa Departemen Informatika, yakni Kerja Praktik (KP).

Kami menyadari masih ada kekurangan baik dalam pelaksanaan kerja praktik maupun penyusunan buku laporan ini. Namun, kami berharap buku laporan ini dapat menambah wawasan pembaca dan dapat menjadi sumber referensi. Kami mengharapkan kritik dan saran yang membangun untuk kesempurnaan buku laporan kerja praktik ini.

Melalui buku ini, kami juga ingin menyampaikan rasa terima kasih kepada orang-orang yang telah membantu, baik langsung maupun tidak langsung, dalam pelaksanaan kerja praktik hingga penyusunan laporan. Orang-orang tersebut antara lain adalah:

1. Kedua orang tua penulis.
2. Bapak Dr.tech.Ir. Raden Venantius Hari Ginardi, M.Sc., selaku dosen pembimbing kerja praktik.
3. Bapak Ary Mazharuddin Shiddiqi, S.Kom., M.Comp.Sc., Ph.D, selaku koordinator Kerja Praktik.
4. Bapak Muhammad Rizki, selaku pembimbing lapangan kami di PT. Qareer Harapan Asia.

Surabaya, Agustus 2020

Raden Bimo Rizki Prayogo

*[Halaman ini sengaja dikosongkan]*

## DAFTAR ISI

|  |    |
|--|----|
| LEMBAR PENGESAHAN .....                  | 5  |
| ABSTRAK.....                             | 8  |
| KATA PENGANTAR .....                     | 9  |
| DAFTAR ISI.....                          | 11 |
| DAFTAR TABEL.....                        | 15 |
| DAFTAR GAMBAR .....                      | 16 |
| DAFTAR KODE .....                        | 17 |
| BAB I PENDAHULUAN.....                   | 19 |
| 1.1. Latar Belakang.....                 | 19 |
| 1.2. Tujuan .....                        | 19 |
| 1.3. Manfaat .....                       | 19 |
| 1.4. Rumusan Masalah.....                | 20 |
| 1.5. Lokasi dan Waktu Kerja Praktik..... | 20 |
| 1.6. Metodologi Kerja Praktik.....       | 20 |
| 1.7. Sistematika Laporan.....            | 21 |
| 1. Bab I Pendahuluan .....               | 21 |
| 2. Bab II Profil Perusahaan .....        | 21 |
| 3. Bab III Tinjauan Pustaka.....         | 22 |
| 4. Bab IV Implementasi .....             | 22 |
| 5. Bab VI Pengujian dan Evaluasi .....   | 22 |
| 6. Bab VII Kesimpulan dan Saran.....     | 22 |
| BAB II PROFIL PERUSAHAAN .....           | 23 |
| 2.1. Profil Perusahaan .....             | 23 |

|                                |   |    |
|--------------------------------|---|----|
| 2.2.                           | Visi dan Misi Perusahaan.....                             | 23 |
| 2.3.                           | Pencapaian Perusahaan .....                               | 23 |
| BAB III TINJAUAN PUSTAKA ..... |   | 25 |
| 3.1.                           | Laravel .....   | 25 |
| 3.2.                           | MySQL .....   | 25 |
| 3.3.                           | Laravel-Excel.....  | 25 |
| BAB IV IMPLEMENTASI .....      |   | 27 |
| 4.1                            | Implementasi fitur <i>Assignment</i> .....                | 27 |
| 4.1.1.                         | Implementasi Model Assignment .....                       | 27 |
| 4.1.2.                         | Implementasi Model AssignmentQuestion .....               | 28 |
| 4.1.3.                         | Implementasi Model AssignmentAnswerSheet .....            | 29 |
| 4.1.4.                         | Implementasi Model AssignmentAnswer .....                 | 31 |
| 4.1.5.                         | Implementasi AssignmentController .....                   | 31 |
| 4.1.6.                         | Implementasi Fungsi Answer.....                           | 35 |
| 4.1.7.                         | Implementasi Fungsi Submit .....                          | 36 |
| 4.1.8.                         | Implementasi AssignmentQuestionController .....           | 37 |
| 4.2                            | Implementasi fitur Laporan Hasil Evaluasi.....            | 39 |
| 4.2.1.                         | Implementasi Fungsi Employee Result.....                  | 39 |
| 4.2.2.                         | Implementasi View Hasil Evaluasi Berdasarkan Individu 40  |    |
| 4.2.3.                         | Implementasi Fungsi Division Result .....                 | 42 |
| 4.2.4.                         | Implementasi View Hasil Evaluasi Berdasarkan Divisi 43    |    |
| 4.2.5.                         | Implementasi Fungsi Skill Set Result .....                | 45 |
| 4.2.6.                         | Implementasi View Hasil Evaluasi Berdasarkan Skill Set 46 |    |

|         |  |    |
|---------|--|----|
| 4.3     | Implementasi fitur Key Performance Index (KPI) Progress.....         | 48 |
| 4.3.1.  | Implementasi Model EmployeeKpiProgress.....                          | 48 |
| 4.3.2.  | Implementasi EmployeeKpiProgressContoller.....                       | 49 |
| 4.3.3.  | Implementasi View Penampilan KPI Progress di User Biasa.....         | 52 |
| 4.3.4.  | Implementasi Fungsi Penampilan Progress KPI pada User Admin.....     | 55 |
| 4.3.5.  | Implementasi View Penampilan Progress KPI pada User Admin.....       | 55 |
| 4.4     | Implementasi fitur Import dan Export .....                           | 58 |
| 4.4.1.  | Implementasi Kelas EmployeeAssignmentExport...                       | 59 |
| 4.4.2.  | Implementasi Fungsi Export Employee Assignment<br>60                 |    |
| 4.4.3.  | Implementasi Kelas EmployeeScoreExport.....                          | 60 |
| 4.4.4.  | Implementasi Fungsi Export Hasil Evaluasi Berdasarkan Individu ..... | 62 |
| 4.4.5.  | Implementasi Kelas DivisionScoreExport .....                         | 63 |
| 4.4.6.  | Implementasi Fungsi Export Hasil Evaluasi Berdasarkan Divisi.....    | 64 |
| 4.4.7.  | Implementasi Kelas SkillSetScoreExport .....                         | 64 |
| 4.4.8.  | Implementasi Fungsi Export Hasil Evaluasi Berdasarkan Skill Set..... | 65 |
| 4.4.9.  | Implementasi Kelas KpiExport.....                                    | 66 |
| 4.4.10. | Implementasi Fungsi Export KPI.....                                  | 67 |
| 4.4.11. | Implementasi Kelas AssignmentImport.....                             | 68 |
| 4.4.12. | Implementasi Fungsi Import Pertanyaan Assignment<br>70               |    |

|                                    |  |    |
|------------------------------------|--|----|
| 4.4.13.                            | Implementasi Kelas EmployeeImport.....   | 71 |
| 4.4.14.                            | Implementasi Fungsi Import Karyawan..... | 74 |
| BAB V PENGUJIAN DAN EVALUASI ..... |  | 75 |
| 5.1.                               | Tujuan Pengujian .....                   | 75 |
| 5.2.                               | Kriteria Pengujian .....                 | 75 |
| 5.3.                               | Skenario Pengujian .....                 | 75 |
| 5.4.                               | Evaluasi Pengujian.....                  | 82 |
| BAB VI KESIMPULAN DAN SARAN .....  |  | 83 |
| 6.1.                               | Kesimpulan .....                         | 83 |
| 6.2.                               | Saran .....                              | 83 |
| DAFTAR PUSTAKA .....               |  | 85 |
| BIODATA PENULIS I.....             |  | 87 |

## **DAFTAR TABEL**

|   |    |
|---|----|
| Tabel 5. 1 Hasil Evaluasi Pengujian ..... | 82 |
|---|----|

## **DAFTAR GAMBAR**

|  |    |
|--|----|
| Gambar 4. 1 Tampilan View Hasil Evaluasi Berdasarkan Individu .....        | 42 |
| Gambar 4. 2 Tampilan View Hasil Evaluasi Berdasarkan Divisi..              | 45 |
| Gambar 4. 3 Tampilan View Hasil Evaluasi Berdasarkan Skill Set .....       | 47 |
| Gambar 4. 4 Tampilan View Penampilan KPI Progress di User Biasa.....       | 54 |
| Gambar 4. 5 Tampilan View Penampilan Progress KPI pada User Admin .....    | 58 |
| <br>Gambar 5. 1 Hasil Penginputan Assignment .....                         | 75 |
| Gambar 5. 2 Hasil Fungsi Answer .....                                      | 76 |
| Gambar 5. 3 Hasil Fungsi Submit.....                                       | 76 |
| Gambar 5. 4 Hasil Laporan Evaluasi Berdasarkan Individu .....              | 76 |
| Gambar 5. 5 Hasil Laporan Evaluasi Berdasarkan Divisi.....                 | 77 |
| Gambar 5. 6 Hasil Laporan Evaluasi Berdasarkan Divisi.....                 | 77 |
| Gambar 5. 7 Hasil Penginputan Progress KPI .....                           | 78 |
| Gambar 5. 8 Hasil Penampilan KPI Progress di Bagian Admin....              | 79 |
| Gambar 5. 9 Hasil Fungsi Export Employee Assignment .....                  | 79 |
| Gambar 5. 10 Hasil Fungsi Export Hasil Evaluasi Berdasarkan Individu ..... | 80 |
| Gambar 5. 11 Hasil Fungsi Export Hasil Evaluasi Berdasarkan Divisi.....    | 80 |
| Gambar 5. 12 Hasil Fungsi Export Hasil Evaluasi Berdasarkan Skill Set..... | 81 |
| Gambar 5. 13 Hasil Fungsi Export KPI .....                                 | 81 |
| Gambar 5. 14 Hasil Fungsi Import Assignment.....                           | 81 |
| Gambar 5. 15 Hasil Fungsi Import Karyawan .....                            | 82 |



## DAFTAR KODE

|   |    |
|---|----|
| Kode 4. 1 Model Assignment .....                                    | 28 |
| Kode 4. 2 Model AssignmentQuestion .....                            | 29 |
| Kode 4. 3 Model AssignmentAnswerSheet .....                         | 30 |
| Kode 4. 4 Model AssignmentAnswer .....                              | 31 |
| Kode 4. 5 Controller Assignment .....                               | 34 |
| Kode 4. 6 Fungsi Answer.....  | 36 |
| Kode 4. 7 Fungsi Submit .....                                       | 37 |
| Kode 4. 8 Controller AssignmentQuestion .....                       | 38 |
| Kode 4. 9 Fungsi Employee Result.....                               | 40 |
| Kode 4. 10 View Hasil Evaluasi Berdasarkan Individu.....            | 41 |
| Kode 4. 11 Fungsi Division Result .....                             | 43 |
| Kode 4. 12 View Hasil Evaluasi Berdasarkan Divisi .....             | 44 |
| Kode 4. 13 View Hasil Evaluasi Berdasarkan Skill Set.....           | 47 |
| Kode 4. 14 Model EmployeeKpiProgress.....                           | 49 |
| Kode 4. 15 Controller EmployeeKpiProgress.....                      | 52 |
| Kode 4. 16 View Penampilan KPI Progress di User Biasa.....          | 53 |
| Kode 4. 17 Fungsi Penampilan Progress KPI pada User Admin...        | 55 |
| Kode 4. 18 View Penampilan Progress KPI pada User Admin .....       | 57 |
| Kode 4. 19 Kelas EmployeeAssignmentExport.....                      | 59 |
| Kode 4. 20 Fungsi Export Employee Assignment .....                  | 60 |
| Kode 4. 21 Kelas EmployeeScoreExport.....                           | 61 |
| Kode 4. 22 Fungsi Export Hasil Evaluasi Berdasarkan Individu...     | 62 |
| Kode 4. 23 Kelas DivisionScoreExport .....                          | 63 |
| Kode 4. 24 Fungsi Export Hasil Evaluasi Berdasarkan Divisi .....    | 64 |
| Kode 4. 25 Kelas SkillSetScoreExport .....                          | 65 |
| Kode 4. 26 Fungsi Export Hasil Evaluasi Berdasarkan Skill Set ..... | 65 |
| Kode 4. 27 Kelas KpiExport.....                                     | 67 |
| Kode 4. 28 Fungsi Export KPI.....                                   | 68 |
| Kode 4. 29 Kelas AssignmentImport.....                              | 69 |
| Kode 4. 30 Fungsi Import Pertanyaan Assignment .....                | 71 |
| Kode 4. 31 Kelas EmployeeImport.....                                | 74 |
| Kode 4. 32 Fungsi Import Karyawan.....                              | 74 |

*[Halaman ini sengaja dikosongkan]*

# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Di zaman Internet ini, berbagai layanan dapat disajikan secara daring untuk meningkatkan kemudahan untuk mengakses layanan tersebut. Salah satu layanan yang dapat disajikan secara daring adalah pelatihan karyawan.

Pada saat ini, Qareer sedang mengembangkan suatu website Learning Management System (LMS) bernama kelas.center untuk memberikan layanan bagi berbagai perusahaan untuk melatih karyawannya secara daring.

Suatu LMS memerlukan berbagai fitur untuk melakukan evaluasi terhadap karyawan yang sedang dilatih dan melakukan manajemen terhadap kinerja karyawan yang sedang dilatih oleh pihak yang berwenang. Proyek Kerja Pratik ini bertujuan mengembangkan berbagai fitur tersebut untuk mempermudah penggunaan website tersebut.

### **1.2. Tujuan**

Tujuan kerja praktik ini adalah untuk menyelesaikan kewajiban kuliah kerja praktik di Institut Teknologi Sepuluh Nopember dengan beban 2 SKS. Serta mengembangkan berbagai fitur manajemen karyawan pada aplikasi LMS kelas.center.

### **1.3. Manfaat**

Manfaat yang dapat diperoleh dengan adanya pengembangan berbagai fitur manajemen karyawan pada aplikasi LMS kelas.center:

- Melakukan evaluasi terhadap karyawan yang sedang dilatih.
- Mempermudah manajemen karyawan oleh pihak yang berwenang.

#### **1.4. Rumusan Masalah**

Berikut ini rumusan masalah pada kerja praktik pengembangan fitur manajemen karyawan:

1. Bagaimana cara menerapkan fitur evaluasi karyawan pada aplikasi LMS kelas.center ?
2. Bagaimana menerapkan fitur manajemen karyawan pada aplikasi LMS kelas.center?

#### **1.5. Lokasi dan Waktu Kerja Praktik**

Mengingat adanya musibah pandemi CoVid-19 pada saat diadakannya kegiatan Kerja Praktik (KP), kegiatan KP ini dilaksanakan secara *Work From Home* atau daring. Kegiatan KP dimulai pada tanggal 22 Juni 2020 hingga 22 Juli 2020.

#### **1.6. Metodologi Kerja Praktik**

Tahapan pengerjaan kerja praktik dapat dijabarkan sebagai berikut:

##### **1. Perumusan Masalah**

Untuk mengetahui fitur-fitur yang perlu diterapkan pada sistem, penulis mengikuti rapat bersama tim developer dari Qareer. Rapat ini diselenggarakan setiap minggu. Sebelum rapat, Chief Technical Officer (CTO) Qareer akan bertemu dengan client yang berminat untuk menggunakan layanan LMS yang sedang dikembangkan. Client akan menguji-coba fitur-fitur yang sudah diimplementasikan dan memberikan saran atau permintaan terhadap suatu fitur. Pada rapat, CTO akan menyampaikan permintaan-permintaan dari client dan tanggapan dari hasil uji coba fitur-fitur oleh client. Lead developer kemudian akan mempertimbangkan hal-hal yang disampaikan CTO dan mengatur prioritas pekerjaan-perkerjaan yang perlu dilaksanakan pada minggu berikutnya. Setelah itu, dilaksanakan pembagian penugasan setiap staff pada tim developer oleh Lead Developer.

## **2. Studi Literatur**

Setelah memahami bagaimana sistem dan environment yang ada pada Qareer, dilakukan studi literatur mengenai cara implementasinya dalam membangun sistem sesuai yang dibutuhkan. Pada tahap ini dilakukan proses pencarian, pembelajaran, pengumpulan dan pemahaman informasi serta literatur yang berkaitan untuk membantu dalam implementasi. Informasi bisa didapat dari internet untuk istilah-istilah umum yang digunakan dalam pengimplementasian suatu sistem.

## **3. Implementasi**

Implementasi didasarkan pada perancangan dan analisis sebelumnya. Kasus penggunaan juga turut mendasari pengimplementasian ini. Pengerjaan dilakukan dengan progres setiap hari, dengan setiap harinya menargetkan perkembangan dari hari sebelumnya.

## **4. Pengujian dan Evaluasi**

Pengujian dilakukan dengan menjalankan berbagai fitur yang diimplementasikan. Kesesuaian data yang dihasilkan saat menjalankan fitur menentukan keberhasilan dalam pengujian.

## **5. Kesimpulan dan Saran**

Bab ini berisi tentang kesimpulan dan saran yang didapatkan dari tugas selama kerja praktik.

### **1.7. Sistematika Laporan**

Laporan KP ini terdiri dari tujuh bab dengan rincian sebagai berikut:

#### **1. Bab I Pendahuluan**

Pada bab ini dijelaskan tentang latar belakang permasalahan, tujuan, waktu pelaksanaan, serta sistematika pengerjaan KP dan juga penulisan laporan KP.

#### **2. Bab II Profil Perusahaan**

Pada bab ini, dijelaskan secara rinci tentang profil perusahaan tempat melaksanakan KP, yakni PT Qareer Harapan Asia.

**3. Bab III Tinjauan Pustaka**

Pada bab ini, dijelaskan mengenai tinjauan pustaka dan literatur yang digunakan dalam penyelesaian KP.

**4. Bab IV Implementasi**

Pada bab ini, berisi penjelasan tahap-tahap yang dilakukan untuk proses implementasi sistem.

**5. Bab VI Pengujian dan Evaluasi**

Pada bab ini, dijelaskan tentang hasil pengujian dan evaluasi dari sistem yang telah dikembangkan selama pelaksanaan KP.

**6. Bab VII Kesimpulan dan Saran**

Pada bab ini, dipaparkan kesimpulan yang dapat diambil dan juga saran selama pengerjaan KP.

## **BAB II**

### **PROFIL PERUSAHAAN**

#### **2.1. Profil Perusahaan**

Qareer Harapan Asia adalah sebuah perusahaan Human Resource berbasis web. Qareer Harapan Asia bertujuan untuk mempermudah pengembangan dan penyaluran sumber daya manusia ke berbagai perusahaan di Asia Tenggara. Qareer Harapan Asia mencapai tujuan tersebut dengan produk-produknya berupa berbagai website pencarian lowongan pekerjaan dan rekrutment untuk perusahaan.

#### **2.2. Visi dan Misi Perusahaan**

Visi dan Misi Qareer Harapan Asia adalah sebagai berikut:

- Visi : Memperdayakan professional dengan wawasan karir
- Misi :
  - Memberikan wawasan, pengetahuan, dan inspirasi bagi para profesional guna meningkatkan kompetensinya
  - Menemukan pasangan yang cocok antara perusahaan dan pencari kerja untuk memaksimalkan kontribusi mereka.
  - Memastikan pencari kerja berkontribusi pada peran yang paling sesuai dengan minat dan keahlian mereka.

#### **2.3. Pencapaian Perusahaan**

Qareer Harapan Asia memperoleh pendanaan untuk salah satu produknya yakni Qerja.com dari SB ISAT Fund dan Emtek Group.

*[Halaman ini sengaja dikosongkan]*



## **BAB III**

### **TINJAUAN PUSTAKA**

Pada bab ini, akan dijelaskan mengenai dasar teori yang digunakan selama proses kerja praktik.

#### **3.1. Laravel**

Laravel adalah framework aplikasi web yang berbasis bahasa PHP. Laravel dimaksudkan untuk pengembangan web dengan arsitektur model - view - controller (MVC) dan berbasis Symfony. Berbagai fitur di laravel bersifat modular sehingga tidak semua fitur akan digunakan, tapi jika ingin menggunakan, dapat memanggil modul tersebut dan dipakai.

#### **3.2. MySQL**

Merupakan salah satu sistem manajemen relasional basis data SQL yang bersifat open source. MySQL digunakan untuk menyimpan data - data yang dapat saling berelasi dengan data yang lain. Untuk melakukan operasi CRUD (Create, Read, Update, Delete) memerlukan query, lalu query tersebut dikirim ke dalam database MySQL. Pada saat query tersebut sampai pada database, database akan mengolah query tersebut dan diterapkan sesuai perintah query.

#### **3.3. Laravel-Excel**

Laravel-Excel adalah sebuah plugin Laravel untuk mempermudah pembuatan dan penginputan data spreadsheet *Microsoft Excel*. Laravel-Excel berbasiskan library *PhpSpreadsheet* yang kemudian dibungkus dengan wrapper ala Laravel untuk mempermudah pengembangan di framework tersebut. Laravel-Excel menawarkan kemudahan untuk melakukan import dan export data spreadsheet dengan sintaks Laravel, peningkatan performa export dengan *automatic chunking* dan pembacaan file excel dengan *chunking*.

*[Halaman ini sengaja dikosongkan]*

## **BAB IV IMPLEMENTASI**

Bab ini membahas tentang implementasi dari perancangan sistem dan pengaplikasian sistem sesuai dengan penugasan yang telah diberikan.

### **4.1 Implementasi fitur *Assignment***

Membangun fitur assignment untuk membuat tugas pada suatu kursus. Tugas tersebut kemudian dapat dikerjakan oleh karyawan yang mengikuti kursus tersebut. Tugas dalam bentuk berupa pertanyaan esai dan jawaban yang diberikan setiap karyawan dapat dilihat pada bagian admin.

Fitur Assignment diimplementasikan dengan beberapa model dan controller.

#### **4.1.1. Implementasi Model Assignment**

##### **4.1.1.1 Deskripsi**

Lapisan model untuk data Assignment.

##### **4.1.1.2 Data**

Model Assignment menyimpan data yang berhubungan dengan detail suatu assignment seperti kursus yang mengadakan assignment tersebut dan batas waktu pengerjaan assignment. Selain itu model ini memiliki one-to-many relationship dengan *AssignmentAnswerSheet* yakni model untuk suatu lembar jawaban seorang karyawan terhadap tersebut assignment dan *AssignmentQuestion* yakni model untuk suatu pertanyaan pada assignment tersebut dan belongs-to relationship dengan *SkillSet* yang merupakan model untuk kursus yang memiliki assignment tersebut.

#### 4.1.1.3 Source Code

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Assignment extends Model
{
    protected $fillable = [
        'skill_set_id', 'timer'
    ];

    public function skill_set()
    {
        return $this->belongsTo('App\SkillSet');
    }

    public function answer_sheets()
    {
        return $this->hasMany('App\AssignmentAnswerSheet');
    }

    public function questions()
    {
        return $this->hasMany('App\AssignmentQuestion');
    }
}
```

*Kode 4. 1 Model Assignment*

### 4.1.2. Implementasi Model AssignmentQuestion

#### 4.1.2.1 Deskripsi

Lapisan model untuk data pertanyaan suatu pertanyaan pada suatu Assignment.

#### 4.1.2.2 Data

Model *AssignmentQuestion* menyimpan data yang berhubungan dengan detail tentang pertanyaan pada suatu assignment seperti isi pertanyaan, gambar pada pertanyaan jika ada. Selain itu model ini memiliki belongs-to relationship dengan *Assignment*.

#### 4.1.2.3 Source Code

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class AssignmentQuestion extends Model
{
    protected $fillable = [
        'question', 'image_question', 'assignment_id'
    ];

    public function assignment()
    {
        return $this->belongsTo('App\Assignment');
    }

    public function answers()
    {
        return $this->hasMany('App\AssignmentAnswer');
    }
}
```

*Kode 4. 2 Model AssignmentQuestion*

### 4.1.3. Implementasi Model AssignmentAnswerSheet

#### 4.1.3.1 Deskripsi

Lapisan model untuk lembar jawaban dari seorang karyawan terhadap pertanyaan-pertanyaan pada suatu Assignment.

#### 4.1.3.2 Data

Model *AssignmentAnswerSheet* menyimpan data yang berhubungan dengan detail tentang lembar jawaban dari seorang karyawan terhadap pertanyaan-pertanyaan pada suatu Assignment seperti karyawan yang mengerjakan dan assignment yang dikerjakan. Selain itu model ini memiliki belongs-to relationship dengan *Assignment* yakni model untuk suatu assignment dan *Employee* yakni model untuk seorang karyawan dan one-to-many relationship dengan *AssignmentAnswer* yang merupakan model untuk jawaban suatu pertanyaan.

#### 4.1.3.3 Source Code

```
namespace App;

use Illuminate\Database\Eloquent\Model;

class AssignmentAnswerSheet extends Model
{
    protected $fillable = [
        'employee_id', 'assignment_id'
    ];

    public function answers()
    {
        return $this->hasMany('App\AssignmentAnswer');
    }

    public function employee()
    {
        return $this->belongsTo('App\Employee');
    }

    public function assignment()
    {
        return $this->belongsTo('App\Assignment');
    }
}
```

Kode 4. 3 Model *AssignmentAnswerSheet*

#### 4.1.4. Implementasi Model AssignmentAnswer

##### 4.1.4.1 Deskripsi

Lapisan model untuk jawaban dari seorang karyawan terhadap pertanyaan pada suatu Assignment.

##### 4.1.4.2 Data

Model *AssignmentAnswer* menyimpan data yang berhubungan dengan detail tentang jawaban dari seorang karyawan terhadap pertanyaan pada suatu Assignment seperti isi jawaban, pertanyaan yang dijawab, dan lembar jawaban karyawan tersebut. Selain itu model ini memiliki one-to-one relationship dengan *AssignmentQuestion* yang merupakan model untuk pertanyaan pada suatu assignment.

##### 4.1.4.3 Source Code

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class AssignmentAnswer extends Model
{
    protected $fillable = [
        'assignment_answer_sheet_id', 'answer', 'image_answer', 'assignment_question_id'
    ];

    public function question()
    {
        return $this->hasOne('App\AssignmentQuestion', 'id', 'assignment_question_id');
    }
}
```

*Kode 4. 4 Model AssignmentAnswer*

#### 4.1.5. Implementasi AssignmentController

##### 4.1.5.1 Deskripsi

Lapisan Controller melakukan kontrol terhadap data yang berhubungan dengan assignment.

#### 4.1.5.2 Data

AssignmentController dapat melaksanakan pembuatan suatu Assignment sekaligus pertanyaannya dalam bentuk AssignmentQuestion. Selain itu, AssignmentController dapat melakukan perubahan terhadap Assignment dan menghapus suatu assignment serta pertanyaannya.

#### 4.1.5.3 Source Code

```
<?php

namespace App\Http\Controllers;
use App\Clients;
use App\Assignment;
use App\AssignmentQuestion;
use App\AssignmentAnswer;
use App\AssignmentAnswerSheet;
use App\Division;
use App\Employee;
use Illuminate\Http\Request;
use App\SkillSet;
use Auth;
use App\EmployeePermission;
use App\EmployeeRolePermission;
use Illuminate\Support\Facades\DB;
use App\Exports\EmployeeAssignmentExport;
use App\Imports\AssignmentImport;
use Maatwebsite\Excel\Facades\Excel;

class AssignmentController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $client = Clients::where('subdomain', session('subdomain'))->first();
        $data = Assignment::with('skill_set')->whereHas('skill_set', function ($q) use ($client)
        {
            $q->where('client_id', $client->id);
        })->get();
        return view('backend.pages.skill_set.assignment.index',[
            'data' => $data
        ]);
    }
}

public static function ProcessImage($image)
{
    $image_question_name = $image->getClientOriginalName();
    $image->move('assets/image_question_assignment/', $image_question_name);
    return $image_question_name;
}
```



```

public function create()
{
    // return $data_ep[0];
    $permission = 'assignment_create';
    $currentRoleID = Auth::guard('internal')->user()->role_id;
    $p = EmployeePermission::where('name',$permission)->first();
    if(!$p)
    {
        dd('permission '. $permission. ' not exists');
    }
    $serp = EmployeeRolePermission::where('role_id',$currentRoleID)->where('employee_permission_id',$p->id)->first();
    if($serp){
        dd('Unauthorized Access, minta tim IT untuk mendapatkan akses');
    }
    $client = Clients::where('subdomain', session('subdomain'))->first();
    return view('backend.pages.skill_set.assignment.create',[
        'skill_set' => SkillSet::where('client_id',$client->id)->get()
    ]);
}

public function store(Request $request)
{
    $permission = 'assignment_create';
    $currentRoleID = Auth::guard('internal')->user()->role_id;
    $p = EmployeePermission::where('name',$permission)->first();
    if(!$p)
    {
        dd('permission '. $permission. ' not exists');
    }
    $serp = EmployeeRolePermission::where('role_id',$currentRoleID)->where('employee_permission_id',$p->id)->first();
    if($serp)
    {
        dd('Unauthorized Access, minta tim IT untuk mendapatkan akses');
    }

    $this->validate($request,[
        'question' => 'required',
        'type_assignment_id' => 'required',
    ]);
    $transaction = function() use ($request)
    {
        $assignment = Assignment::firstOrCreate([
            'skill_set_id' => $request->type_assignment_id
        ]);

        if ($assignment->timer != $request->timer) {
            $assignment->timer = $request->timer;
            $assignment->save();
        }

        foreach($request->question as $key => $value){
            $data = [
                'assignment_id' => $assignment->id,
                'question' => $value
            ];

            if ($request->file('image_question') && array_key_exists($key,$request->file('image_question')) {
                $image = self::ProcessImage($request->file('image_question')[$key]);
                $data['image_question'] = $image;
            }

            $question = AssignmentQuestion::create($data);
            if (!$question) {
                return back()->with('danger','Assignment Gagal Diinputkan');
            }
        }

        return back()->with('success','Assignment Berhasil Diinputkan');
    };
}

```

```

public function show(Assignment $assignment)
{
    $assignment_name = $assignment->skill_set->skill_set_name;
    $data = $assignment->questions;

    return view('backend.pages.skill_set.assignment.show',[
        'assignment_name' => $assignment_name,
        'data'=> $data
    ]);
}

public function update(Request $request, Assignment $assignment)
{
    $this->validate($request,[
        'timer' => 'integer'
    ]);
    if ($request->timer) {
        $assignment->timer = $request->timer;
        $assignment->save();
        return back()->with('success','Assignment Berhasil Diedit');
    }
}

public function destroy(Assignment $assignment)
{
    foreach ($assignment->questions as $question) {
        $question->answers()->delete();
    }
    $assignment->questions()->delete();
    $assignment->answer_sheets()->delete();
    $assignment->delete();
    return response()->json([
        'message'=>'Data Deleted Successfully'
    ]);
}

```

*Kode 4. 5 Controller Assignment*

## 4.1.6. Implementasi Fungsi Answer

### 4.1.6.1 Deskripsi

Fungsi answer adalah fungsi yang akan membuat suatu lembar jawaban untuk karyawan yang akan mengerjakan suatu assignment. Fungsi ini berada di *AssignmentController*.

### 4.1.6.2 Parameter

Fungsi answer menerima ID assignment yang akan dikerjakan.

### 4.1.6.3 Data

Fungsi answer akan membuat lembar jawaban untuk karyawan yang akan mengerjakan suatu assignment dalam bentuk *AssignmentAnswerSheet*. Karyawan tersebut kemudian akan diarahkan ke halaman untuk mengerjakan assignment tersebut.

### 4.1.6.4 Source Code

```
public function answer($id)
{
    $assignment = Assignment::find($id);
    if (!$assignment) {
        return back()->with('danger','Unexpected Error');
    }
    $uid = Auth::guard('user')->user()->id;

    $answer_sheet = AssignmentAnswerSheet::where('assignment_id', $assignment->id)
        ->where('employee_id', $uid)
        ->first();
}
```

```

if(!$answer_sheet)
{
    $answer_sheet = AssignmentAnswerSheet::create(
        [
            'assignment_id' => $assignment->id,
            'employee_id' => $uid
        ]
    );

    if (!$answer_sheet) {
        return back()->with('danger', 'Unexpected Error');
    }
}
else if ($answer_sheet->done){
    return back()->with('danger', 'Anda sudah mengerjakan assignment');
}

return view('frontend.pages.assignment.show',
[
    'data' => $answer_sheet
]);

```

*Kode 4. 6 Fungsi Answer*

## 4.1.7. Implementasi Fungsi Submit

### 4.1.7.1 Deskripsi

Fungsi submit adalah fungsi yang akan menerima ID dari lembar jawaban yang menjadi tempat pengerjaan assignment. Fungsi ini berada di *AssignmentController*.

### 4.1.7.2 Parameter

Fungsi submit menerima ID lembar jawaban yang digunakan.

### 4.1.7.3 Data

Fungsi submit akan membuat jawaban dari karyawan dalam bentuk *AssignmentAnswer* dan menghubungkannya ke lembar jawaban atau *AssignmentAnswerSheet* yang menjadi tempat pengerjaan assignment tersebut. Karyawan tersebut kemudian akan diarahkan ke halaman utama kursus.

#### 4.1.7.4 Source Code

```
public function submit(Request $request, $id)
{
    $uid = Auth::guard('user')->user()->id;
    if ($uid != $request->uid) {
        return back()->with('danger','Assignment Gagal Diinputkan');
    }
    $answer_sheet = AssignmentAnswerSheet::where('id', $id)
                                           ->where('employee_id', $uid)
                                           ->first();

    if (!$answer_sheet)
    {
        return back()->with('danger','Assignment Gagal Diinputkan');
    }

    $transaction = function() use ($request, $answer_sheet)
    {
        foreach($request->answer as $key => $value)
        {
            $data = [
                'assignment_answer_sheet_id' => $answer_sheet->id,
                'answer' => $value? $value: "-",
                'assignment_question_id' => $key
            ];
            $answer = AssignmentAnswer::create($data);
            if (!$answer) {
                return back()->with('danger','Assignment Gagal Diinputkan');
            }
        }
        $answer_sheet->done = true;
        $answer_sheet->save();

        return redirect('user/home')->with('success','Assignment Berhasil Diinputkan');
    };

    $return = DB::transaction($transaction);
    return $return;
}
```

*Kode 4. 7 Fungsi Submit*

### 4.1.8. Implementasi AssignmentQuestionController

#### 4.1.8.1 Deskripsi

Lapisan Controller melakukan kontrol terhadap data pertanyaan suatu assignment.

#### 4.1.8.2 Data

AssignmentController dapat melakukan perubahan dan penghapusan terhadap AssignmentQuestion

#### 4.1.8.3 Source Code

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\AssignmentQuestion;

class AssignmentQuestionController extends Controller
{
    public static function ProcessImage($image)
    {
        $image_question_name = $image->getClientOriginalName();
        $image->move('assets/image_question_assignment/', $image_question_name);
        return $image_question_name;
    }

    public function update(Request $request, $id)
    {
        $this->validate($request, [
            'question' => 'string',
            'image_question' => 'image'
        ]);
        $question = AssignmentQuestion::find($id);
        $data = array_filter(['question' => $request->question != $question->question ? $request->question : null ]);
        if ($request->file('image_question')) {
            $image = self::ProcessImage($request->file('image_question'));
            $data['image_question'] = $image;
        }
        if ($data) {
            $question->update($data);
        }

        return back()->with('success', 'Pertanyaan Berhasil Diedit');
    }

    public function destroy( $id)
    {
        $question = AssignmentQuestion::find($id);
        $question->answers()->delete();
        $question->delete();
        return response()->json([
            'message' => 'Data Deleted Successfully'
        ]);
    }
}
```

*Kode 4. 8 Controller AssignmentQuestion*

## **4.2 Implementasi fitur Laporan Hasil Evaluasi**

Membangun fitur laporan hasil evaluasi untuk menampilkan hasil evaluasi dari setiap karyawan pada bagian admin. Evaluasi adalah suatu tes yang dilaksanakan setelah karyawan menyelesaikan semua materi yang ada pada suatu kursus. Hasil evaluasi tersebut kemudian dapat dilihat oleh karyawan yang berwenang, seperti admin, dan ditampilkan berdasarkan individual, divisi pada suatu perusahaan atau berdasarkan kursus yang ada.

Fitur laporan diimplementasikan sebagai beberapa fungsi dan *view* atau antarmuka.

### **4.2.1. Implementasi Fungsi Employee Result**

#### **4.2.1.1 Deskripsi**

Fungsi untuk mengquery hasil evaluasi suatu kursus secara individual.

#### **4.2.1.2 Parameter**

Fungsi `employee_result` tidak memiliki parameter.

#### **4.2.1.3 Data**

Fungsi ini mengquery setiap karyawan pada suatu perusahaan dan melakukan join terhadap jabatannya pada suatu divisi dan kursus yang diperlukan untuk jabatan tersebut. Data tersebut kemudian diteruskan ke *view* yang akan menampilkan hasil evaluasi setiap individu.

#### **4.2.1.4 Source Code**

```

public function employee_result()
{
    $client = Clients::where('subdomain', session('subdomain'))->first();

    $data = Employee::where(function($q){
        return $q->whereNotNull('division_id')
            ->orWhereNotNull('job_level_id')
            ->orWhereNotNull('division_info_id')
            ->orWhere('division_id', '<>',0)
            ->orWhere('job_level_id', '<>',0)
            ->orWhere('division_info_id', '<>',0);
    })->where('client_id',$client->id)->with('division_info','division_info.skill_set')->get();

    return view('backend.pages.result.employee',[
        'data' => $data,
        'skillset' => SkillSet::where('client_id',$client->id)->get(),
        'division' => Division::where('client_id',$client->id)->get(),
    ]);
}

```

*Kode 4. 9 Fungsi Employee Result*

## 4.2.2. Implementasi View Hasil Evaluasi Berdasarkan Individu

### 4.2.2.1 Deskripsi

View untuk menampilkan hasil evaluasi suatu kursus secara individual.

### 4.2.2.2 Data

View ini menampilkan nilai persentase, ranking berdasarkan divisi dan ranking berdasarkan kursus seorang pada suatu evaluasi. Nilai tersebut dalam bentuk suatu progress bar yang akan berubah warna jika nilai evaluasi kurang dari atau lebih dari rata-rata nilai pada divisi yang ditunjukkan pada progress bar dibawahnya. Jika nilai evaluasi tersebut sama dengan atau lebih besar dari rata-rata maka warna progress bar nilai akan menjadi hijau, jika kurang dari rata-rata maka warnanya akan berubah menjadi merah.



### 4.2.2.3 Source Code

```

<table id="example" class="table table-bordered ">
  <thead class="text-center">
    <tr>
      <td scope="col" rowspan="2">Employee Name</td>
      <td scope="col" rowspan="2">Position</td>
      <td scope="col" rowspan="2">Skill Set</td>
      <td scope="col" colspan="2">Rank </td>
      <td scope="col" rowspan="2">Score</td>
    </tr>
    <tr>
      <td>In Division</td>
      <td>In Skill Set</td>
    </tr>
  </thead>
  <tbody>
    @if(count($data) == 0)
      <tr>
        <td colspan="6" class="text-center">No Data Found!</td>
      </tr>
    @elseif(count($data) > 0)

@foreach($data as $employee)

  @php
    $div = $employee->division
    ->Join('employees', 'employees.division_id', '-', 'divisions.id')
    ->where('division_id', $employee->division_id)->get();
    $employees_div_id = $div? $div->pluck('id')->all():0;
    $total_div = $div? count($employees_div_id):0;

  @endphp
  @foreach ($employee->division_info->skill_set as $skill_set)
    <tr>
      @php
        $ss = $skill_set->pivot
        ->Join('employees', 'employees.division_info_id', '-', 'division_info_skill_set.division_info_id')
        ->where('skill_set_id', $skill_set->id)->get();
        $employees_ss_id = $ss? $ss->pluck('id')->all():0;
        $total_ss = $ss? count($employees_ss_id):0;
        $quiz = $skill_set->quiz->first();
        $score = $quiz? $quiz->completion->where('user_id', $employee->id)->where('type_quiz', 2)->pluck('score')->first():0;
        $avg = $quiz? $quiz->completion->whereIn('user_id', $employees_div_id)->where('type_quiz', 2)->avg('score') :0;
        $rank_div = $quiz && $score? $quiz->completion->whereIn('user_id', $employees_div_id)
        ->where('score', '>', $score)->where('type_quiz', 2)->count() + 1:$total_div;
        $rank_ss = $quiz && $score? $quiz->completion->whereIn('user_id', $employees_ss_id)
        ->where('score', '>', $score)->where('type_quiz', 2)->count() + 1:$total_ss;

      @endphp
      <td> {{ $employee->employee_name }} </td>
      <td> {{ $employee->division->division_name }} {{ $employee->job_level->name }} </td>
      <td> {{ $skill_set->skill_set_name }} </td>
      <td> {{ $rank_div }} / {{ $total_div }} </td>
      <td> {{ $rank_ss }} / {{ $total_ss }} </td>
      <td>
        <div class="progress">
          <div class="progress-bar" role="progressbar"
            style="width:{{($score? $score:0)}}%;background-color:{{($score >= $avg? '#04993f': '#d6a1b1')}};{{($score > 15 ? "" : 'color:black;')}} !important"
            aria-valuemin="15" aria-valuemax="100">{{($score? round($score,2):0)}}</div>
          </div>
          <div class="progress" style="height: 8px">
            <div class="progress-bar" role="progressbar" style="width:{{($avg? $avg:0)}}%;{{($avg > 15 ? "" : 'color:black;')}}"
              aria-valuemin="15" aria-valuemax="100">{{($avg? round($avg,2):0)}}</div>
          </div>
        </td>
      </tr>
    @endforeach
  @endforeach

```

Kode 4. 10 View Hasil Evaluasi Berdasarkan Individu

4.2.2.4 Tampilan yang Dihasilkan

| Employee Name  | Position    | Skill Set        | Rank        |              | Score                     |
|----------------|-------------|------------------|-------------|--------------|---------------------------|
|                |             |                  | In Division | In Skill Set |                           |
| ABDUL ROSID    |             |                  |             |              |                           |
| HRD Staff      |             |                  |             |              |                           |
| ABDUL ROSID    | HRD Staff   | OnBoarding Skill | 1 / 219     | 1 / 219      | <div><div></div>100</div> |
| ACHMAD RABBANI |             |                  |             |              |                           |
| HRD Manager    |             |                  |             |              |                           |
| ACHMAD RABBANI | HRD Manager | OnBoarding Skill | 219 / 219   | 219 / 219    | <div><div></div>5</div>   |
| ACHMAD RABBANI | HRD Manager | Leadership Skill | 219 / 219   | 17 / 17      | <div><div></div>0</div>   |
| ACHMAD SOFYAN  |             |                  |             |              |                           |
| HRD Staff      |             |                  |             |              |                           |
| ACHMAD SOFYAN  | HRD Staff   | OnBoarding Skill | 3 / 219     | 3 / 219      | <div><div></div>0</div>   |

Gambar 4. 1 Tampilan View Hasil Evaluasi Berdasarkan Individu

4.2.3. Implementasi Fungsi Division Result

4.2.3.1 Deskripsi

Fungsi untuk mengquery hasil evaluasi suatu kursus berdasarkan divisi.

4.2.3.2 Parameter

Fungsi division\_result tidak memiliki parameter.

4.2.3.3 Data

Fungsi ini mengquery setiap divisi pada suatu perusahaan dan melakukan join terhadap jabatan pada divisi tersebut. Data tersebut kemudian diteruskan ke view yang akan menampilkan hasil evaluasi setiap divisi.

#### 4.2.3.4 Source Code

```
public function division_result()
{
    $permission = 'division_read';
    $_currentRoleID = Auth::guard('internal')->user()->role_id;
    $_p = EmployeePermission::where('name',$permission)->first();
    if(!$p)
    {
        dd('permission '. $permission. ' not exists');
    }
    $erp = EmployeeRolePermission::where('role_id',$_currentRoleID)->where('employee_permission_id',$_p->id)->first();
    if($erp){
    }else{
        return '<h3>Unauthorized Access, minta tim IT untuk mendapatkan akses</h3>';
    }
    $client = Clients::where('subdomain', session('subdomain'))->first();
    $data = Division::where('client_id',$client->id)->with(['division_info'])->get();

    return view('backend.pages.result.division',['data' => $data]);
}
```

*Kode 4. 11 Fungsi Division Result*

#### 4.2.4. Implementasi View Hasil Evaluasi Berdasarkan Divisi

##### 4.2.4.1 Deskripsi

View untuk menampilkan hasil evaluasi suatu kursus berdasarkan divisi.

##### 4.2.4.2 Data

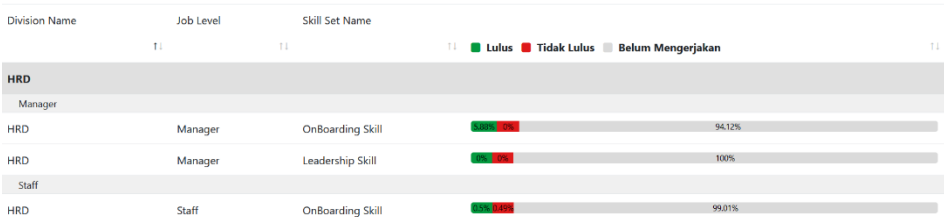
View ini menampilkan persentase karyawan yang lulus, yang tidak lulus dan yang belum mengerjakan suatu evaluasi pada suatu kursus.

#### 4.2.4.3 Source Code

[illegible]

*Kode 4. 12 View Hasil Evaluasi Berdasarkan Divisi*

### 4.2.4.4 Tampilan yang Dihasilkan



Gambar 4. 2 Tampilan View Hasil Evaluasi Berdasarkan Divisi

### 4.2.5. Implementasi Fungsi Skill Set Result

#### 4.2.5.1 Deskripsi

Fungsi untuk mengquery hasil evaluasi suatu kursus berdasarkan divisi.

#### 4.2.5.2 Parameter

Fungsi skill\_set\_result tidak memiliki parameter.

#### 4.2.5.3 Data

Fungsi ini mengquery setiap kursus pada suatu perusahaan dan melakukan join terhadap jabatan pada tiap divisi. Data tersebut kemudian diteruskan ke view yang akan menampilkan hasil evaluasi setiap divisi berdasarkan skill set.

#### 4.2.5.4 Source Code

```
public function skill_set_result()
{
    $client = Clients::where('subdomain', session('subdomain'))->first();
    $data = SkillSet::where('client_id', $client->id)->with('division_info')->get();
    return view('backend.pages.result.skillset', ['data' => $data]);
}
```



```

@foreach($data as $skill_set)
@foreach ($skill_set->division_info as $division_info)
    @php
        $employees = $division_info->employees->pluck('id')->val();
        $no_emp = $employees ? count($employees) : 1;
        $quiz = $skill_set->quiz->first();
        $completion = $quiz? $quiz->completion->where('type_quiz',2)->whereIn('user_id', $employees) :0;
        $avg = $completion? $completion->avg('score') :0;
        $completed = $completion ? round(($completion->count() * 100 / $no_emp),2) :0;

        $pass = $completion? round(($completion->where('score','>=',$avg)->count() * 100 / $no_emp),2) :0;
        $fail = $completion? round(($completed - $pass),2) :0;
        $not_done = round((100 - $completed),2) ;

    @endphp
    <tr>
    <td>
        {{ $skill_set->skill_set_name }}
    </td>
    <td>
        {{ $division_info->job_level->name }}
    </td>
    <td>
        {{ $division_info->division->division_name }}
    </td>
    <td>
        <div class="progress">
            <div class="progress-bar" role="progressbar" style="width: {{($pass < 5 ? 5 : $pass)}}%;background-color:#04993f;{{($pass > 15 ? "" : "color:black;")); !important"
                aria-valuemin="15" aria-valuemax="100">{{round($pass,2)}}%</div>
            <div class="progress-bar" role="progressbar" style="width: {{($fail < 5 ? 5 : $fail)}}%;background-color:#d44b3b;{{($fail > 15 ? "" : "color:black;")); !important"
                aria-valuemin="30" aria-valuemax="100">{{round($fail,2)}}%</div>
            <div class="progress-bar" role="progressbar"
                style="width: {{($not_done < 5 ? 5 : $not_done)}}%;background-color:#dadada;{{($not_done > 15 ? "color:black;" : "")}} !important;"
                aria-valuemin="30" aria-valuemax="100">{{round($not_done,2)}}%</div>
        </div>
    </td>
    </tr>
    @endforeach
    @endif
</tbody>
</table>

```

Kode 4. 13 View Hasil Evaluasi Berdasarkan Skill Set

## 4.2.6.4 Tampilan yang Dihasilkan

| Skill Set Name                                   | Job Level | Division Name |  |        |
|--|-----------|---------------|--|--------|
|  |           |               | <div> <div></div> <div></div> <div></div> </div> |        |
| <div> <div></div> <div></div> <div></div> </div> |           |               |  |        |
| Leadership Skill                                 |           |               |  |        |
| Leadership Skill                                 | Manager   | HRD           | <div> <div></div> <div></div> <div></div> </div> | 100%   |
| OnBoarding Skill                                 |           |               |  |        |
| OnBoarding Skill                                 | Manager   | HRD           | <div> <div></div> <div></div> <div></div> </div> | 94.12% |
| OnBoarding Skill                                 | Staff     | HRD           | <div> <div></div> <div></div> <div></div> </div> | 99.01% |

Gambar 4. 3 Tampilan View Hasil Evaluasi Berdasarkan Skill Set

### **4.3 Implementasi fitur Key Performance Index (KPI) Progress**

Key Performance Index (KPI) adalah ukuran pencapaian suatu divisi pada suatu perusahaan. Fitur yang diimplementasikan berfungsi untuk menginputkan progress seorang karyawan terhadap KPI divisinya. Selain penginputan data progress, statistik progress dapat dilihat pada bagian admin maupun user biasa.

Fitur laporan diimplementasikan sebagai beberapa model, fcontroller dan *view* atau antarmuka.

#### **4.3.1. Implementasi Model EmployeeKpiProgress**

##### **4.3.1.1 Deskripsi**

Lapisan model untuk data progress KPI seorang karyawan.

##### **4.3.1.2 Data**

Model *EmployeeKpiProgress* menyimpan data yang berhubungan dengan detail tentang progress KPI seorang karyawan seperti karyawan yang mengerjakan KPI, KPI yang dikerjakan, tanggal progress dilaksanakan, hari progress dilaksanakan, progress yang dikerjakan pada hari tersebut dan akumulasi progress yang dikerjakan. Adapula fungsi query untuk mendapatkan total progress dan fungsi untuk menyelipkan query where pada tanggal hari ini. Selain itu model ini memiliki belongs-to relationship dengan model *Kpi* yang merupakan model untuk data KPI.



### 4.3.1.3 Source Code

```
<?php

namespace App;
use Carbon\Carbon;

use Illuminate\Database\Eloquent\Model;

class EmployeeKpiProgress extends Model
{
    protected $fillable = [
        'employee_id', 'kpi_id', 'date', 'count_days', 'todays_progress', 'total_progress'
    ];

    public function kpi()
    {
        return $this->belongsTo('App\Kpi');
    }

    public function scopeToday($query)
    {
        return $query->where('date', 'like', Carbon::today(+7)->toDateString());
    }

    public function scopeTotal($query)
    {
        return $query->max('total_progress');
    }

    public function scopeFindKpiUser($query, $kpi, $user)
    {
        return $query->where(['kpi_id'=> $kpi , 'employee_id'=> $user]);
    }
}
```

*Kode 4. 14 Model EmployeeKpiProgress*

## 4.3.2. Implementasi EmployeeKpiProgressContoller

### 4.3.2.1 Deskripsi

Lapisan Controller melakukan kontrol terhadap data KPI progress seorang karyawan.

### 4.3.2.2 Data

EmployeeKpiProgressController dapat melaksanakan pembuatan suatu progress KPI dalam bentuk EmployeeKpiProgress. Selain itu, AssignmentContoller dapat mengquery statistik progress KPI seorang karyawan dan menruskannya ke sebuah view untuk menampilkan data tersebut.

### 4.3.2.3 Source Code

```
namespace App\Http\Controllers;

use App\Clients;
use DB;
use App\EmployeeKpiProgress;
use App\Kpi;
Use Auth;
use Illuminate\Http\Request;
use Carbon\Carbon;

class EmployeeKpiProgressController extends Controller
{
    public function show($kpi_id)
    {
        $client = Clients::where('subdomain', session('subdomain'))->first();
        $find = EmployeeKpiProgress::findKpiUser($kpi_id,Auth::guard('user')->user()->id->count('id');

        if (!$find) {
            $k = Kpi::find($kpi_id)->with('progress')->whereHas('progress',function ($query)
            {
                $query->where('employee_id', Auth::guard('user')->user()->id);
            })->first();
        }
        $k = Kpi::where('client_id',$client->id)->with('progress')->find($kpi_id);

        $today = $find ? $k->progress()->today()->first():false;
        $current = $find ? $k->progress()->total() :false;

        $max = $find ? $k->progress()->orderBy('todays_progress','desc')->first() :false;
        $min = $find ? $k->progress()->orderBy('todays_progress', 'asc')->first() :false;
        $avg = $current ? $current / $k->progress()->count('id') : 0;
        $date = [];
        $vals = [];
        if ($find) {
            $date = $k->progress()->orderBy('date','asc')->pluck('date')->toArray();
            $vals = $k->progress()->orderBy('date','asc')->pluck('todays_progress')->toArray();
        }
    }
}
```

```

        return view('frontend.pages.progress.index',[
            'kpi' => $k,
            'today' => $today ? $today->today_progress : 0,
            'current' => $current ? $current : 0,
            'max' => $max ? $max : false,
            'min' => $min ? $min : false,
            'avg' => $avg,
            'date' => $date,
            'vals' => $vals
        ]);
    }

    public function upsert(Request $request, $kpi)
    {
        $this->validate($request,[
            'date' => 'required',
            'progress' => 'required'
        ]);
        $kpi_up = Kpi::find($kpi);
        if (!$kpi_up) {
            return back()->with('Fail','Progress gagal diinputkan');
        }

        $transaction = function () use ($kpi_up, $request)
        {
            $user = Auth::guard('user')->user()->id;
            $date = new Carbon($request->date);
            $day = $date->toArray()['day'];

            $progress = $request->progress;
            $kpi_up->progress = $kpi_up->progress + $request->progress;
            $q1 = $kpi_up->save();

            if (!$q1) {
                return back()->with('Fail','Progress gagal diinputkan');
            }

            $q2 = null;

            $today = EmployeeKpiProgress::findKpiUser($kpi_up->id,$user)->today()->first();
            $total = EmployeeKpiProgress::findKpiUser($kpi_up->id,$user)->first()? EmployeeKpiProgress::findKpiUser($kpi_up->id,$user)->total() : false;
            if ($today) {
                $today->today_progress = $today->today_progress + $progress;
                $today->total_progress = $total + $progress;
                $q2 = $today->save();
            } else {
                $q2 = EmployeeKpiProgress::create(
                    [
                        'kpi_id'=> $kpi_up->id,
                        'employee_id'=> $user,
                        'date' => $date->toDateString(),
                        'count_days' => $day,
                        'today_progress' => $progress,
                        'total_progress' => $total? $total + $progress : $progress
                    ]
                );
            }
            if (!$q2) {
                return back()->with('Fail','Progress gagal diinputkan');
            }

            return back()->with('success','Progress berhasil diinputkan');
        };
    }

```

```
$return = DB::transaction($transaction);
return $return;
```

*Kode 4. 15 Controller EmployeeKpiProgress*

### 4.3.3. Implementasi View Penampilan KPI Progress di User Biasa

#### 4.3.3.1 Deskripsi

View ini statistik progress KPI untuk user biasa.

#### 4.3.3.2 Data

View ini menampilkan statistik progress KPI seorang karyawan.

#### 4.3.3.3 Source Code

```
<div class="container">
  <div class="col-md-12">
    <div class="panel-deck">
      <div class="panel" style="background-color: #1f2123;">
        <div class="panel-body">
          <h3>{{ $kpi->xname }}</h3>
          @php
            $total = $kpi->target;
            $percent = ($current/$total) * 100;
            $unit = $kpi->unit;
          @endphp
          <h5 class="panel-title white">Progress Anda <b style="color: #c19b70;">{{(number_format($today))}} {{ $unit }}</b> hari ini</h5>
          <small class="text-muted">Progress</small>
          <p class="panel-text white">{{(number_format($current))}}/{{(number_format($total))}} {{ $unit }}</p>
          <div class="progress">
            <div class="progress-bar" id="cust" role="progressbar" aria-valuenow="" aria-valuemin="0" aria-valuemax="100"
              style="width:{{ $percent > 100 ? 100 : $percent }}%;height:20px;background-color:#c19b70;{{ $percent > 2 ? "" : "color:black;" }}">
              {{ round ($percent,2) }}%
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
  <br><br>
  <div class="col-md-12">
    <div class="panel-deck">
      <div class="panel" style="background-color: #1f2123;">
        <div class="panel-body">
          <h5 class="panel-title white">Statistik</h5>
          <small class="text-muted">Rata-rata per hari</small>
          <p class="panel-text white"><b>{{(number_format($avg))}} </b> {{ $unit }}</p>
          <canvas id="myChart"></canvas>
        </div>
      </div>
    </div>
  </div>
</div>
<br><br>
```

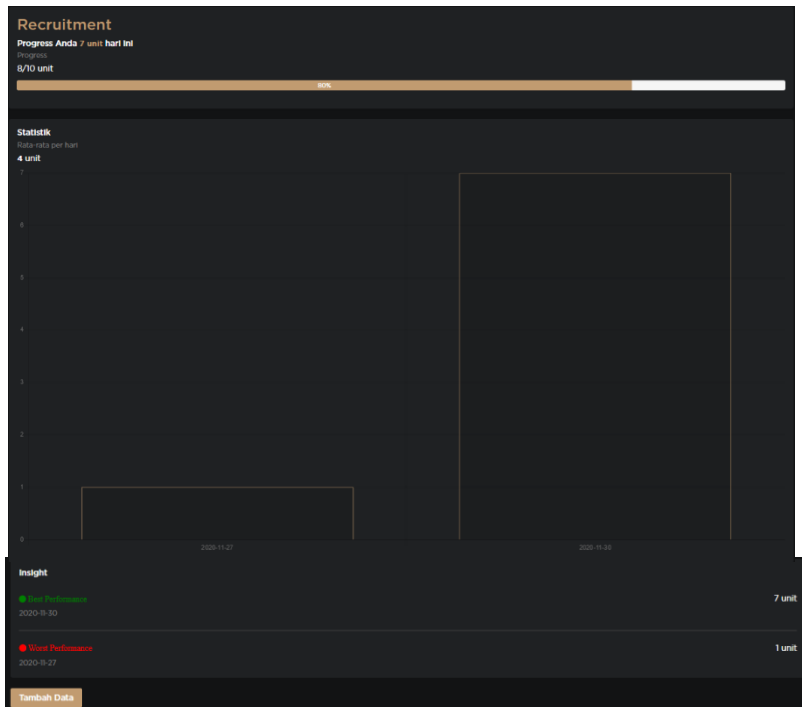
```

<div class="col-md-12">
  <div class="panel-deck">
    <div class="panel" style="background-color: #1f2123;">
      <div class="panel-body">
        <h5 class="panel-title white">Insight</h5>
        <br>
        <i class="fa fa-circle" style="color: green;"> Best Performance</i>
        <p class="pull-right white">{{ $max? number_format($max->today's_progress) : 0 }} {{ $unit }} </p>
        <br>
        <small class="text-muted">{{ $max? $max->date : 'Belum ada progress' }} </small>
        <br><br>
        <i class="fa fa-circle" style="color: red;"> Worst Performance</i>
        <p class="pull-right white">{{ $min? number_format($min->today's_progress) : 0 }} {{ $unit }}</p>
        <br>
        <small class="text-muted">{{ $min? $min->date : 'Belum ada progress' }} </small>
      </div>
    </div>
  </div>
  <button class="btn btn-default" data-toggle="modal" data-target="#myModal">Tambah Data</button>
  <div class="modal fade" id="myModal" tabindex="-1" role="dialog" aria-labelledby="myModalLabel" aria-hidden="true">
    <div class="modal-dialog">
      <div class="modal-content">
        <div class="modal-header">
          <button type="button" class="close" data-dismiss="modal"><span aria-hidden="true">x</span><span class="sr-only">Close</span></button>
          <h4 class="modal-title" id="myModalLabel"><i class="fa fa-plus-square-o"></i> Masukkan Progres Hari Ini</h4>
        </div>
        <div class="modal-body">
          <form action="{{ route('progress.upsert', $kpi->id) }}" method="post">
            @csrf
            <div class="input-group">
              <small class="text-muted">Tanggal</small>
              <input type="date" class="form-control" name="date">
              <small class="text-muted">Progress Hari Ini ({{ $unit }})</small>
              <input type="text" class="form-control" name="progress">
              <span class="input-group-btn">
                <button class="btn btn-default" type="submit">
                  data-toggle="tooltip" data-placement="button">
                    Tambahkan
                </button>
              </span>
            </div>
          </form>
        </div>
      </div>
    </div>
  </div>
</div>
</div>

```

*Kode 4. 16 View Penampilan KPI Progress di User Biasa*

### 4.3.3.4 Tampilan yang Dihasilkan



Masukkan Progres Hari Ini

Tanggal

dd / mm / yyyy

Progress Hari Ini (unit)

Tambahkan

Gambar 4. 4 Tampilan View Penampilan KPI Progress di User Biasa

#### 4.3.4. Implementasi Fungsi Penampilan Progress KPI pada User Admin

##### 4.3.4.1 Deskripsi

Fungsi ini untuk mengquery terhadap progress KPI pada suatu divisi yang melaksanakan KPI tersebut. Fungsi ini berada pada bagian admin.

##### 4.3.4.2 Parameter

Fungsi ini memiliki parameter berupa ID KPI yang ditampilkan.

##### 4.3.4.3 Data

Fungsi ini melakukan query terhadap progress KPI dalam bentuk EmployeeKpiProgress yang dikerjakan oleh setiap karyawan pada suatu divisi yang melaksanakan KPI tersebut. Data yang diquery kemudian diteruskan ke view yang menampilkan progress setiap karyawan terhadap KPI tersebut

##### 4.3.4.4 Source Code

```
public function show(Kpi $kpi)
{
    $client = Clients::where('subdomain', session('subdomain'))->first();
    $emp = Employee::where(['division_id'=>$kpi->division_id,'job_level_id'=>$kpi->job_level_id,'client_id' => $client->id])->with(['division','job_level'])->get();
    return view('backend.pages.kpi.show',[
        'emp' => $emp,
        'kpi' => $kpi
    ]);
}
```

*Kode 4. 17 Fungsi Penampilan Progress KPI pada User Admin*

#### 4.3.5. Implementasi View Penampilan Progress KPI pada User Admin

##### 4.3.5.1 Deskripsi

View ini menampilkan progress KPI pada suatu divisi yang melaksanakan KPI tersebut. Fungsi ini berada pada bagian admin.

### 4.3.5.2 Data

View ini menampilkan progress KPI yang dikerjakan oleh setiap karyawan pada suatu divisi yang melaksanakan KPI tersebut. Jika nama karyawan diklik maka akan muncul modal untuk menampilkan statistic progress KPI karyawan tersebut secara detail.

### 4.3.5.3 Source Code

```
<table class="table table-bordered table-responsive-md table-responsive-lg table-responsive-xl">
  <thead style="background-color: #fafa" >
    <tr>
      <td>No</td>
      <td>Employee Name</td>
      <td>Division</td>
      <td>Job Level</td>
      <td>Progress</td>
    </tr>
  </thead>
  <tbody>
    @if (count($emp) == 0)
      <tr>
        <td colspan="5" class="text-center">No Data Found</td>
      </tr>
    @elseif(count($emp) > 0)
      @foreach ($emp as $d)
        @php
          $unit = $kpi->unit;
          $total = $kpi->target;
          $max = $d->progress()->where('kpi_id',$kpi->id)->first() ? $d->progress()->where('kpi_id',$kpi->id)->orderBy('today_progress','desc')->first() : 0;
          $min = $d->progress()->where('kpi_id',$kpi->id)->first() ? $d->progress()->where('kpi_id',$kpi->id)->orderBy('today_progress','asc')->first() : 0;
          $avg = $d->progress()->where('kpi_id',$kpi->id)->first() ? $d->progress()->where('kpi_id',$kpi->id)->avg('today_progress') : 0;
          $current = $d->progress()->where('kpi_id',$kpi->id)->first() ? $d->progress()->where('kpi_id',$kpi->id)->total() : 0;
          $percent = ($current/$total) * 100;

        @endphp
        <tr>
          <td>{{ $loop->index+1 }}</td>
          <td>{{ $d->name }}</td>
          <td>{{ $d->division_name }}</td>
          <td>{{ $d->job_level_name }}</td>
          <td>
            @if (count($d->nomor_karyawan) > 0)
              <div class="progress">
                <div class="progress-bar" id="cust" role="progressbar" aria-valuenow="" aria-valuemin="0" aria-valuemax="100" style="width:{{ $percent > 100 ? 100 : $percent }}%;height:20px">
                </div>
              </div>
            @endif
          </td>
        </tr>
      @endforeach
    @endif
  </tbody>
</table>
```



```

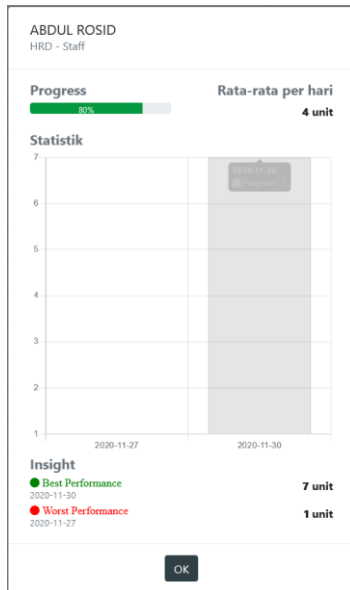
<div class="modal fade" id="employee{{ $d->id }}" tabindex="-1" role="dialog" aria-labelledby="exampleModallabel" aria-hidden="true">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <div class="col">
          <h5 class="modal-title">{{ $d->employee_name }}</h5>
          <h6 class="text-muted">{{ $d->division->division_name }} - {{ $d->job_level->name }}</h6>
        </div>
      </div>
      <div class="modal-body">
        <div class="container">
          <div class="row">
            <div class="col">
              <h5 class="font-weight-bold text-muted">Progress</h5>
              <div class="progress">
                <div class="progress-bar" id="cust" role="progressbar" aria-valuenow="" aria-valuemin="0" aria-valuemax="100"
                  style="width:{{ $percent > 100 ? 100 : $percent }}%;height:20px;background-color:#04993f;{{ $percent > 15 ? "" : "color:black;" }}">
                  {{ round ($percent,2) }}%
                </div>
              </div>
            </div>
            <div class="col text-right">
              <h5 class="font-weight-bold text-muted">Rata-rata per hari</h5>
              <p class="font-weight-bold"><b>{{ number_format($avg) }} </b> {{ $unit }}</p>
            </div>
          </div>
          <h5 class="font-weight-bold text-muted">Statistik</h5>
          <div>
            <canvas id="watching-chart{{ $d->id }}" class="chart" width="200" height="200"></canvas>
          </div>
          <h5 class="font-weight-bold text-muted">Insight</h5>
          <div class="row">
            <div class="col">
              <i class="fa fa-circle" style="color: ■green;"> Best Performance</i>
              <br>
              <small class="text-muted">{{ $max? $max->date : 'Belum ada progress' }} </small>
            </div>
            <div class="col">
              <p class="font-weight-bold text-right"> <b>{{ $max? number_format($max->today's_progress) : 0 }} </b> {{ $unit }}</p>
            </div>
          </div>
          <div class="row">
            <div class="col">
              <i class="fa fa-circle" style="color: ■red;"> Worst Performance</i>
              <br>
              <small class="text-muted">{{ $min? $min->date : 'Belum ada progress' }} </small>
            </div>
            <div class="col">
              <p class="font-weight-bold text-right"> <b>{{ $min? number_format($min->today's_progress) : 0 }} </b> {{ $unit }}</p>
            </div>
          </div>
        </div>
      </div>
      <div class="modal-footer justify-content-center">
        <button type="button" class="btn text-white " data-dismiss="modal" style="background-color: ■#2f3f44;">
          OK
        </button>
      </div>
    </div>
  </div>
</div>

```

*Kode 4. 18 View Penampilan Progress KPI pada User Admin*

#### 4.3.5.4 Tampilan yang Dihasilkan

| No | Employee Name            | Division | Job Level | Progress |
|----|--------------------------|----------|-----------|----------|
| 1  | EFLINA HARAHAP           | HRD      | Staff     | 0%       |
| 2  | SURYADI                  | HRD      | Staff     | 0%       |
| 3  | LILIS NURYANI            | HRD      | Staff     | 0%       |
| 4  | MUHAMMAD HILMY OCTAWIANO | HRD      | Staff     | 0%       |



Gambar 4. 5 Tampilan View Penampilan Progress KPI pada User Admin

#### 4.4 Implementasi fitur Import dan Export

Membuat fitur import dan export untuk menginputkan berbagai data dengan file excel dan mengouputkan berbagai data sebagai file excel. Fitur import dan export diimplementasikan dengan plugin Laravel-Excel untuk mempermudah dan meningkatkan performa import dan export data excel.

Untuk mengimplementasi import atau export suatu data, kita perlu membuat kelas Import atau Export sesuai data yang

diinginkan terlebih dahulu dan membuat fungsi untuk melakukan import atau export sesuai pendefinisian kelas tersebut.

#### 4.4.1. Implementasi Kelas EmployeeAssignmentExport

##### 4.4.1.1 Deskripsi

Kelas Export untuk melakukan export terhadap data suatu lembar jawaban assignment yang dimiliki seorang karyawan.

##### 4.4.1.2 Parameter

Kelas Export memiliki parameter berupa ID lembar jawaban yang ingin diexport

##### 4.4.1.3 Data

Kelas EmployeeAssignmentExport mengquery data lembar jawaban assignment dalam bentuk AssignmentAnswerSheet dan melakukan join terhadap data assignment, jawaban assignment dan karyawan.

##### 4.4.1.4 Source Code

```
<?php

namespace App\Exports;

use App\AssignmentAnswerSheet;
use Illuminate\Contracts\View\View;
use Maatwebsite\Excel\Concerns\FromView;

class EmployeeAssignmentExport implements FromView
{
    protected $id;

    public function __construct( $id)
    {
        $this->id = $id;
    }

    public function view(): View
    {
        $data = AssignmentAnswerSheet::with('assignment','answers','employee')->find($this->id);
        return view('backend.exports.assignment.employee', [
            'data' => $data
        ]);
    }
}
```

*Kode 4. 19 Kelas EmployeeAssignmentExport*

## 4.4.2. Implementasi Fungsi Export Employee Assignment

### 4.4.2.1 Deskripsi

Fungsi ini akan mengexport data lembar jawaban dalam bentuk excel.

### 4.4.2.2 Parameter

Fungsi ini memiliki parameter berupa ID lembar jawaban yang ingin diexport

### 4.4.2.3 Data

Fungsi `employee_export` akan membuat file excel data lembar jawaban yang diinginkan sesuai yang didefinisikan pada kelas `EmployeeAssignmentExport`.

### 4.4.2.4 Source Code

```
public function employee_export($id)
{
    return Excel::download(new EmployeeAssignmentExport($id),
        'answer_sheet-' . $id . '.xlsx');
}
```

*Kode 4. 20 Fungsi Export Employee Assignment*

## 4.4.3. Implementasi Kelas EmployeeScoreExport

### 4.4.3.1 Deskripsi

Kelas Export untuk melakukan export terhadap data suatu hasil evaluasi dari seorang karyawan.

### 4.4.3.2 Parameter

Kelas Export memiliki parameter berupa filter hasil evaluasi yang ingin diexport.

#### 4.4.3.3 Data

Kelas `EmployeeScoreExport` mengquery data karyawan dalam bentuk `Employee` sesuai filter yang diinginkan dan melakukan join terhadap data jabatan karyawan pada divisi dan kursus. Data filter mencakup kursus dan divisi yang ingin diekspor. Data query tersebut kemudian diteruskan ke view yang akan menjadi format dari hasil export. View yang digunakan sama dengan view untuk Laporan Hasil Evaluasi berdasarkan Karyawan.

#### 4.4.3.4 Source Code

```
<?php

namespace App\Exports;

use App\Clients;
use App\Employee;
use Illuminate\Contracts\View\View;
use Maatwebsite\Excel\Concerns\FromView;

class EmployeeScoreExport implements FromView
{
    protected $filter;

    public function __construct(array $filter)
    {
        $this->filter = $filter;
    }

    public function view(): View
    {
        $client = Clients::where('subdomain', session('subdomain'))->first();
        $whereDivision = array_filter([
            'client_id' => $client->id,
            'division_id' => $this->filter['division_id'],
        ]);
        $whereSkillSet = array_filter([
            'client_id' => $client->id,
            'skill_sets.id' => $this->filter['skill_set_id'],
        ]);

        $data = Employee::where(function($q){
            return $q
                ->orWhereNotNull('job_level_id')
                ->orWhereNotNull('division_info_id')
                ->orWhere('job_level_id', '<>',0)
                ->orWhere('division_info_id', '<>',0);
        })->where($whereDivision)->with('division_info','division_info.skill_set')
        ->whereHas('division_info.skill_set', function($q) use ($whereSkillSet) {
            return $q
                ->where($whereSkillSet);
        })->get();

        return view('backend.exports.results.employee', [
            'data' => $data,
            'skill_set_id' => array_key_exists('skill_sets.id',$whereSkillSet) ? $whereSkillSet['skill_sets.id'] : null
        ]);
    }
}
```

*Kode 4. 21 Kelas `EmployeeScoreExport`*

#### 4.4.4. Implementasi Fungsi Export Hasil Evaluasi Berdasarkan Individu

##### 4.4.4.1 Deskripsi

Fungsi untuk melakukan export terhadap data hasil evaluasi beberapa karyawan.

##### 4.4.4.2 Parameter

Fungsi ini memiliki parameter berupa filter hasil evaluasi yang ingin diexport.

##### 4.4.4.3 Data

Fungsi export akan menerima filter yang diinginkan dan membuat file excel hasil evaluasi yang diinginkan sesuai dengan filter yang diinginkan dan query yang didefinisikan pada kelas EmployeeScoreExport.

##### 4.4.4.4 Source Code

```
public function export(Request $request)
{
    $filter = [
        'division_id' => $request->division,
        'skill_set_id' => $request->skillset,
    ];
    $name = 'employees';
    $name = $request->division ?$name.'-div-'. $request->division : $name;
    $name = $request->skillset ?$name.'-skillset-'. $request->skillset : $name;
    $name = $request->periode ?$name.'-periode-'. $request->periode : $name;
    $name = $name.'.xlsx';

    return Excel::download(new EmployeeScoreExport($filter),$name);
}
```

*Kode 4. 22 Fungsi Export Hasil Evaluasi Berdasarkan Individu*

## 4.4.5. Implementasi Kelas DivisionScoreExport

### 4.4.5.1 Deskripsi

Kelas Export untuk melakukan export terhadap data suatu hasil evaluasi dari seorang karyawan.

### 4.4.5.2 Parameter

Kelas Export memiliki parameter berupa filter hasil evaluasi yang ingin diexport.

### 4.4.5.3 Data

Kelas DivisionScoreExport mengquery data divisi dalam bentuk Division dan melakukan join terhadap data jabatan yang ada pada divisi tersebut. Data query tersebut kemudian diteruskan ke view yang akan menjadi format dari hasil export. View yang digunakan sama dengan view untuk Laporan Hasil Evaluasi berdasarkan divisi.

### 4.4.5.4 Source Code

```
<?php

namespace App\Exports;

use App\Clients;
use App\Division;
use Illuminate\Contracts\View\View;
use Maatwebsite\Excel\Concerns\FromView;

class DivisionScoreExport implements FromView
{
    public function view(): View
    {
        {
            $client = Clients::where('subdomain', session('subdomain'))->first();
            $data = Division::where('client_id', $client->id)->with(['division_info'])->get();
            return view('backend.exports.results.division', [
                'data' => $data
            ]);
        }
    }
}
```

*Kode 4. 23 Kelas DivisionScoreExport*

#### **4.4.6. Implementasi Fungsi Export Hasil Evaluasi Berdasarkan Divisi**

##### **4.4.6.1 Deskripsi**

Kelas Export untuk melakukan export terhadap data hasil evaluasi beberapa karyawan.

##### **4.4.6.2 Data**

Fungsi export akan membuat file excel hasil evaluasi yang diinginkan sesuai query yang didefinisikan pada kelas DivisionScoreExport.

##### **4.4.6.3 Source Code**

```
public function export()  
{  
    return Excel::download(new DivisionScoreExport, 'divisions.xlsx');  
}
```

*Kode 4. 24 Fungsi Export Hasil Evaluasi Berdasarkan Divisi*

#### **4.4.7. Implementasi Kelas SkillSetScoreExport**

##### **4.4.7.1 Deskripsi**

Kelas Export untuk melakukan export terhadap data suatu hasil evaluasi dari seorang karyawan.

##### **4.4.7.2 Data**

Kelas SkillSetScoreExport mengquery data kursus dalam bentuk SkillSet dan melakukan join terhadap data jabatan yang ada pada divisi tersebut. Data query tersebut kemudian diteruskan ke view yang akan menjadi format dari hasil export. View yang digunakan sama dengan view untuk Laporan Hasil Evaluasi berdasarkan Skill Set.



### 4.4.7.3 Source Code

```
<?php

namespace App\Exports;

use App\Clients;
use App\SkillSet;
use Illuminate\Contracts\View\View;
use Maatwebsite\Excel\Concerns\FromView;

class SkillSetScoreExport implements FromView
{
    public function view(): View
    {
        $client = Clients::where('subdomain', session('subdomain'))->first();
        $data = SkillSet::where('client_id', $client->id)->with('division_info')->get();

        return view('backend.exports.results.skillset', [
            'data' => $data
        ]);
    }
}
```

*Kode 4. 25 Kelas SkillSetScoreExport*

## 4.4.8. Implementasi Fungsi Export Hasil Evaluasi Berdasarkan Skill Set

### 4.4.8.1 Deskripsi

Kelas Export untuk melakukan export terhadap data hasil evaluasi berdasarkan kursus.

### 4.4.8.2 Data

Fungsi export akan membuat file excel hasil evaluasi yang diinginkan sesuai query yang didefinisikan pada kelas SkillSetScoreExport.

### 4.4.8.3 Source Code

```
public function export()
{
    return Excel::download(new SkillSetScoreExport, 'skillsets.xlsx');
}
```

*Kode 4. 26 Fungsi Export Hasil Evaluasi Berdasarkan Skill Set*

## 4.4.9. Implementasi Kelas KpiExport

### 4.4.9.1 Deskripsi

Kelas Export untuk melakukan export terhadap data KPI suatu divisi.

### 4.4.9.2 Parameter

Kelas Export ini memiliki parameter berupa filter KPI diexport.

### 4.4.9.3 Data

Kelas KpiExport mengquery data KPI sesuai filter yang diinginkan dan melakukan join terhadap data divisi, jabatan divisi tersebut dan progress KPI. Data filter mencakup divisi, jabatan pada divisi dan periode bulan yang ingin diexport.

### 4.4.9.4 Source Code

```
<?php

namespace App\Exports;

use App\Clients;
use App\EmployeeKpiProgress;
use DB;
use Illuminate\Contracts\View\View;
use Maatwebsite\Excel\Concerns\FromView;

class KpiExports implements FromView
{
    protected $filter;

    public function __construct(array $filter)
    {
        $this->filter = $filter;
    }
}
```

```

public function view(): View
{
    $client = Clients::where('subdomain', session('subdomain'))->first();
    $where = array_filter([
        'kpi.client_id' => $client->id,
        'division_id' => $this->filter['division_id'],
        'job_level_id' => $this->filter['job_level_id'],
        'month_period' => $this->filter['month_period'],
    ]);

    $data = DB::table('kpis')
        ->join('divisions', 'divisions.id', '=', 'kpis.division_id')
        ->join('job_levels', 'job_levels.id', '=', 'kpis.job_level_id')
        ->where($where)
        ->select('kpis.*', 'divisions.division_name', 'job_levels.name as job_level_name')
        ->get();

    $ids = $data->pluck('id')->all();
    $sum = EmployeeKpiProgress::selectRaw('kpi_id, employee_id, SUM(todays_progress) sum')
        ->whereIn('kpi_id', $ids)
        ->groupBy('kpi_id')
        ->groupBy('employee_id')
        ->get();
    $avg = $sum->groupBy('kpi_id')->map(function ($row) {
        return $row->avg('sum');
    });

    return view('backend.exports.kpi', [
        'data' => $data,
        'avg' => $avg->all()
    ]);
}
}

```

*Kode 4. 27 Kelas KpiExport*

## 4.4.10. Implementasi Fungsi Export KPI

### 4.4.10.1 Deskripsi

Fungsi Export untuk melakukan export terhadap data KPI.

### 4.4.10.2 Parameter

Fungsi ini memiliki parameter berupa filter KPI ingin diexport.

#### 4.4.10.3 Data

Fungsi export akan menerima filter yang diinginkan dan membuat file excel hasil evaluasi yang diinginkan sesuai dengan filter yang diinginkan dan query yang didefinisikan pada kelas KpiExport.

#### 4.4.10.4 Source Code

```
public function kpi_export(Request $request)
{
    $filter = [
        'division_id' => $request->division,
        'job_level_id' => $request->job_level,
        'month_period' => $request->periode,
    ];
    $name = 'kpi';
    $name = $request->division ?$name.'-div-'. $request->division : $name;
    $name = $request->job_level ?$name.'-job_level-'. $request->job_level : $name;
    $name = $request->periode ?$name.'-periode-'. $request->periode : $name;
    $name = $name.'.xlsx';

    return Excel::download(new KpiExports($filter),$name);
}
```

*Kode 4. 28 Fungsi Export KPI*

### 4.4.11. Implementasi Kelas AssignmentImport

#### 4.4.11.1 Deskripsi

Kelas Import untuk melakukan export terhadap data suatu pertanyaan suatu assignment.

#### 4.4.11.2 Data

Kelas AssignmentImport membuat data pertanyaan assignment dalam bentuk AssignmentQuestion. Proses import melakukan validasi terhadap kolom question yang merupakan isi dari pertanyaan. Baris yang menghasilkan eror akan di langkahi. Selain itu, proses import dibaca dengan batch dengan ukuran

seratus dan chunk dengan ukuran seratus untuk mengurangi penggunaan RAM saat pembacaan.

### 4.4.11.3 Source Code

```
<?php

namespace App\Imports;

use App\AssignmentQuestion;

use Maatwebsite\Excel\Concerns\SkipsOnFailure;
use Maatwebsite\Excel\Concerns\SkipsOnError;
use Maatwebsite\Excel\Concerns\SkipsErrors;
use Maatwebsite\Excel\Concerns\SkipsFailures;
use Maatwebsite\Excel\Concerns\Importable;
use Maatwebsite\Excel\Concerns\WithValidation;
use Maatwebsite\Excel\Concerns\ToModel;
use Maatwebsite\Excel\Concerns\WithHeadingRow;
use Maatwebsite\Excel\Concerns\WithChunkReading;
use Maatwebsite\Excel\Concerns\WithBatchInserts;

class AssignmentImport implements ToModel, WithHeadingRow, WithValidation ,WithBatchInserts, WithChunkReading, SkipsOnFailure, SkipsOnError
{
    use Importable, SkipsFailures, SkipsErrors;

    /**
     * @param array $row
     *
     * @return \Illuminate\Database\Eloquent\Model|null
     */
    protected $data;

    public function __construct($data)
    {
        $this->data = $data;
    }

    public function model(array $row)
    {
        $assignment_id = $this->data;
        return new AssignmentQuestion([
            'assignment_id' => $assignment_id,
            'question' => $row['question'],
            'image_question' => $row['image_question'],
        ]);
    }

    public function rules(): array
    {
        return [
            'question' => 'required',
        ];
    }

    public function batchSize(): int
    {
        return 100;
    }

    public function chunkSize(): int
    {
        return 100;
    }
}
```

*Kode 4. 29 Kelas AssignmentImport*

## 4.4.12. Implementasi Fungsi Import Pertanyaan Assignment

### 4.4.12.1 Deskripsi

Kelas Import untuk melakukan import terhadap data pertanyaan suatu assignment.

### 4.4.12.2 Parameter

Kelas Import menerima parameter berupa data assignment dan file excel yang berisi pertanyaan assignment tersebut.

### 4.4.12.3 Data

Fungsi import akan menginputkan informasi tentang assignment yang akan dibuat. Informasi assignment yang dibutuhkan berupa batas waktu dan kursus dari assignment tersebut. Fungsi ini kemudian akan membaca file excel yang berisi pertanyaan assignment tersebut dan menginputkan data tersebut ke database sesuai yang didefinisikan pada kelas AssignmentImport.

### 4.4.12.4 Source Code

```
public function import(Request $request)
{
    $this->validate($request,[
        'timer' => 'required',
        'type_assignment_id' => 'required'
    ]);

    $assignment = Assignment::firstOrCreate([
        'skill_set_id' => $request->type_assignment_id
    ]);

    if ($assignment->timer != $request->timer) {
        $assignment->timer = $request->timer;
        $assignment->save();
    }
}
```

```

        if($request->file('image_question'))
        {
            foreach ($request->file('image_question') as $image) {
                self::ProcessImage($image, 'pretest_question');
            }
        }

        $data = $assignment->id;
        $import = new AssignmentImport($data);
        $import->import($request->file('file'));
        if ($import->failures()->isNotEmpty()) {
            return back()->withFailures($import->failures());
        }

        return back()->with('success', 'Pertanyaan Berhasil Diinputkan');
    }
}

```

*Kode 4. 30 Fungsi Import Pertanyaan Assignment*

#### **4.4.13. Implementasi Kelas EmployeeImport**

##### **4.4.13.1 Deskripsi**

Kelas Import untuk melakukan export terhadap data karyawan.

##### **4.4.13.2 Parameter**

Kelas Import ini menerima parameter berupa data assignment dan file excel yang berisi pertanyaan assignment tersebut.

##### **4.4.13.3 Data**

Kelas EmployeeImport membuat data karyawan berupa divisi, jabatan, role, password dan informasi biodata lainnya dalam bentuk model Employee. Kelas ini pertama-tama akan membaca nama divisi, jabatan dan role suatu karyawan. Divisi, jabatan dan role tersebut kemudian akan diquery untuk mendapatkan masing IDnya. Divisi, jabatan dan role yang sudah diquery kemudian akan disimpan ke cache untuk mempermudah dan mempercepat pengisian informasi karyawan lainnya.

Proses import melakukan validasi terhadap kolom nama, email, nama divisi, nama jabatan dan nama role suatu karyawan. Baris yang menghasilkan eror akan dilangkahi. Selain itu, proses import dibaca dengan batch dengan ukuran lima-ratus dan chunk dengan ukuran lima-ratus untuk mengurangi penggunaan RAM saat pembacaan.

#### 4.4.13.4 Source Code

```
<?php

namespace App\Imports;

use App\Employee;
use App\JobLevel;
use App\EmployeeRole;
use App\Division;
use App\DivisionInfo;
use App\Clients;
use Illuminate\Support\Facades\Hash;
use Maatwebsite\Excel\Concerns\SkipsOnFailure;
use Maatwebsite\Excel\Concerns\SkipsOnError;
use Maatwebsite\Excel\Concerns\SkipsFailures;
use Maatwebsite\Excel\Concerns\Importable;
use Maatwebsite\Excel\Concerns\WithValidation;
use Maatwebsite\Excel\Concerns\ToModel;
use Maatwebsite\Excel\Concerns\WithChunkReading;
use Maatwebsite\Excel\Concerns\WithHeadingRow;
use Maatwebsite\Excel\Concerns\WithBatchInserts;

class EmployeeImport implements ToModel, WithHeadingRow, WithValidation, WithBatchInserts, WithChunkReading, SkipsOnFailure, SkipsOnError
{
    use Importable, SkipsFailures, SkipsErrors;
    /**
     * @param array $row
     *
     * @return \Illuminate\Database\Eloquent\Model|null
     */
    protected $cache;

    public function __construct()
    {
        $this->cache = [];
    }

    public function model(array $row)
    {
        if (array_key_exists('client',$this->cache)) {
            $client = Clients::where('subdomain', session('subdomain'))->first();
            $this->cache['client'] = $client->id;
        }
        $client = $this->cache['client'];
        if (array_key_exists($row['division_name'].'_d',$this->cache)) {
            $division = Division::where("division_name", $row['division_name'])->firstOr(function () use ($row, $client)
            {
                return Division::create([
                    'division_name' => $row['division_name'],
                    'client_id' => $client
                ]);
            });
            $this->cache[$row['division_name'].'_d'] = $division->id;
        }
        if (array_key_exists($row['job_level_name'].'_j',$this->cache)) {
            $job_level = JobLevel::where("name", $row['job_level_name'])->firstOr(function () use ($row, $client)
            {
                return JobLevel::create([
                    'name' => $row['job_level_name'],
                    'client_id' => $client
                ]);
            });
            $this->cache[$row['job_level_name'].'_j'] = $job_level->id;
        }
    }
}
```



```

if (!array_key_exists($row['role_name']."_r",$this->cache)) {
    $employee_role = EmployeeRole::where("name",$row['role_name'])->firstOr(function () use ($row)
    {
        return EmployeeRole::create([
            'name' => $row['role_name']
        ]);
    });
    $this->cache[$row['role_name']."_r"] = $employee_role->id;
}

if (!array_key_exists($row['division_name']."_".$row['job_level_name'],$this->cache)) {
    $division_info = DivisionInfo::firstOrCreate([
        'job_level_id' => $this->cache[$row['job_level_name']."_j"],
        'division_id' => $this->cache[$row['division_name']."_d"],
    ],
    [
        'job_level_id' => $this->cache[$row['job_level_name']."_j"],
        'division_id' => $this->cache[$row['division_name']."_d"],
        'client_id' => $this->cache['client']
    ]);
    $this->cache[$row['division_name']."_".$row['job_level_name']] = $division_info->id;
}

if (!array_key_exists('password',$this->cache)) {
    $this->cache['password'] = Hash::make('123123');
}

return new Employee([
    'employee_name' => $row['employee_name'],
    'email' => $row['email'],
    'username' => $row['username'],
    'password' => $this->cache['password'],
    'phone' => $row['phone'],
    'client_id' => $this->cache['client'],
    'division_id' => $this->cache[$row['division_name']."_d"],
    'division_info_id' => $this->cache[$row['division_name']."_".$row['job_level_name']],
    'job_level_id' => $this->cache[$row['job_level_name']."_j"],
    'role_id' => $this->cache[$row['role_name']."_r"]
]);

public function rules(): array
{
    return [
        'employee_name' => 'required',
        '*.email' => 'required|email|unique:employees,email',
        'division_name' => 'required',
        'job_level_name' => 'required',
        'role_name' => 'required',
    ];
}

```

```

public function batchSize(): int
{
    return 500;
}

public function chunkSize(): int
{
    return 500;
}

```

*Kode 4. 31 Kelas EmployeeImport*

## 4.4.14. Implementasi Fungsi Import Karyawan

### 4.4.14.1 Deskripsi

Kelas Import untuk melakukan import terhadap data pertanyaan suatu assignment.

### 4.4.14.2 Data

Fungsi ini akan membaca file excel yang bersisi data karyawan tersebut dan menginputkan data tersebut ke database sesuai yang didefinisikan pada kelas EmployeeImport.

### 4.4.14.3 Source Code

```

public function import(Request $request)
{
    $this->validate($request,[
        'file' => 'required',
    ],[
        'file.required' => 'File is Required',
    ]);

    $import = new EmployeeImport;
    $import->import($request->file('file'));
    if ($import->failures()->isNotEmpty()) {
        return back()->withFailures($import->failures());
    }
    return redirect()->route('employee.index');
}

```

*Kode 4. 32 Fungsi Import Karyawan*

## BAB V

### PENGUJIAN DAN EVALUASI

Bab ini menjelaskan tahap uji coba dilakukan terhadap Sistem yang ada dan penugasan yang telah dilakukan. Pengujian dilakukan untuk memastikan kualitas yang dibangun dan kesesuaian hasil dari fitur yang diimplementasi dengan kebutuhan yang direncanakan.

#### 5.1. Tujuan Pengujian

Pengujian dilakukan terhadap penugasan yang diterima dan dikerjakan.

#### 5.2. Kriteria Pengujian

Penilaian atas pencapaian tujuan pengujian didapatkan dengan memerhatikan beberapa hasil yang diharapkan berikut ini:

- Kesesuaian dalam memenuhi kebutuhan yang telah didefinisikan perusahaan.
- Kesesuaian data yang didapatkan atau dikirimkan dari fungsi pemanggilan API.
- Tidak adanya kesalahan dari sistem yang telah dibuat.

#### 5.3. Skenario Pengujian

Skenario pengujian dilakukan dengan menjalankan fitur yang diimplementasi dan melihat apakah fitur sudah berjalan sesuai dengan kebutuhan yang diinginkan.

##### 5.3.1. Fitur Assignment




##### 5.3.1.1. Hasil Penginputan Assignment

Show Assignment Leadership Skill

Back

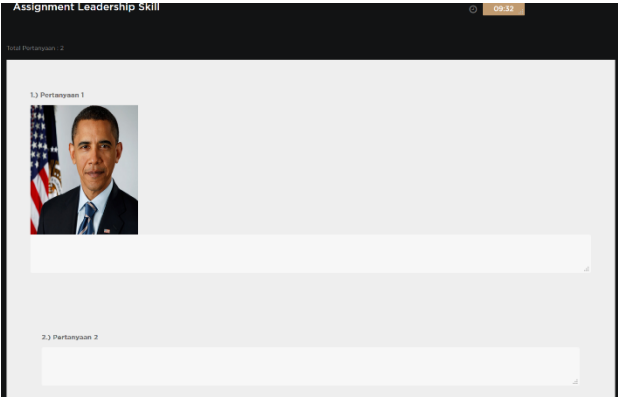
Show 10 entries

Search:

| No | T1 | Pertanyaan   | T1 | Gambar  | T1 | Action  | T1 |
|----|----|--------------|----|---|----|---|----|
| 1  |    | Pertanyaan 1 |    |  |    |   |    |
| 2  |    | Pertanyaan 2 |    |   |    |   |    |

*Gambar 5. 1 Hasil Penginputan Assignment*

5.3.1.2. Hasil Fungsi Answer




Gambar 5. 2 Hasil Fungsi Answer

5.3.1.3. Hasil Fungsi Submit

Employees Assignment Progress

Show10Entities

Search

| Employee Name  | Division | Job Level | Skill Set        | Finish Date | Action  |
|----------------|----------|-----------|------------------|-------------|---|
| ABDUL ROSID    |          |           |                  |             |   |
| HRD Staff      |          |           |                  |             |   |
| ABDUL ROSID    | HRD      | Staff     | Onboarding Skill |             |   |
| ACHMAD RABBANI |          |           |                  |             |   |
| HRD Manager    |          |           |                  |             |   |
| ACHMAD RABBANI | HRD      | Manager   | Onboarding Skill |             |   |
| ACHMAD RABBANI | HRD      | Manager   | Leadership Skill | 2020-12-09  |  |

Gambar 5. 3 Hasil Fungsi Submit

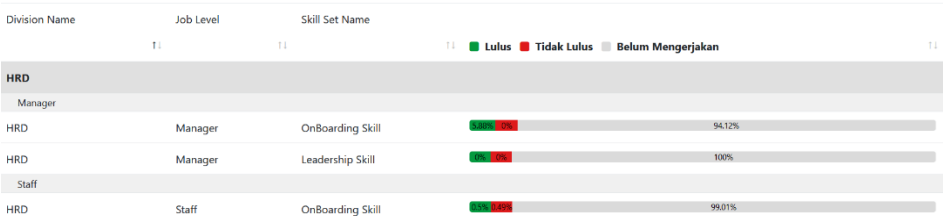
5.3.2. Fitur Laporan Hasil Evaluasi

5.3.2.1. Hasil Laporan Evaluasi Berdasarkan Individu

| Employee Name  | Position    | Skill Set        | Rank        |              | Score                                    |
|----------------|-------------|------------------|-------------|--------------|--|
|                |             |                  | In Division | In Skill Set |  |
| ABDUL ROSID    |             |                  |             |              |  |
| HRD Staff      |             |                  |             |              |  |
| ABDUL ROSID    | HRD Staff   | Onboarding Skill | 1 / 219     | 1 / 219      | <div><div>100</div><div>100%</div></div> |
| ACHMAD RABBANI |             |                  |             |              |  |
| HRD Manager    |             |                  |             |              |  |
| ACHMAD RABBANI | HRD Manager | Onboarding Skill | 219 / 219   | 219 / 219    | <div><div>0</div><div>0%</div></div>     |
| ACHMAD RABBANI | HRD Manager | Leadership Skill | 219 / 219   | 17 / 17      | <div><div>0</div><div>0%</div></div>     |
| ACHMAD SOFYAN  |             |                  |             |              |  |
| HRD Staff      |             |                  |             |              |  |
| ACHMAD SOFYAN  | HRD Staff   | Onboarding Skill | 3 / 219     | 3 / 219      | <div><div>50</div><div>50%</div></div>   |

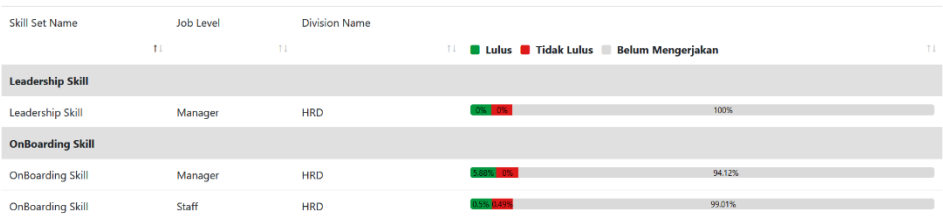
Gambar 5. 4 Hasil Laporan Evaluasi Berdasarkan Individu

5.3.2.2. Hasil Laporan Evaluasi Berdasarkan Divisi



Gambar 5. 5 Hasil Laporan Evaluasi Berdasarkan Divisi

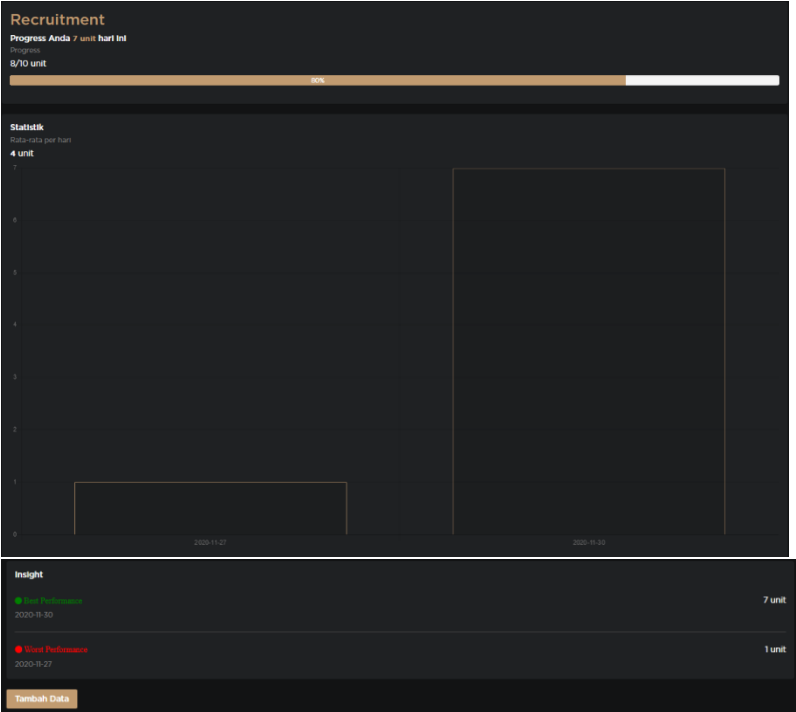
5.3.2.3. Hasil Laporan Evaluasi Berdasarkan Divisi



Gambar 5. 6 Hasil Laporan Evaluasi Berdasarkan Divisi

5.3.3.Fitur Progress KPI

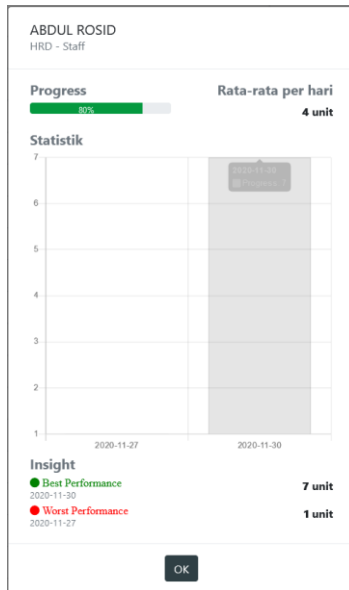
5.3.3.1. Hasil Penginputan Progress KPI



Gambar 5. 7 Hasil Penginputan Progress KPI

5.3.3.2. Hasil Penampilan KPI Progress di Bagian Admin








| No | Employee Name            | Division | Job Level | Progress       |
|----|--------------------------|----------|-----------|----------------|
| 1  | EFLINA HARAHAP           | HRD      | Staff     | 0% <div></div> |
| 2  | SURWADI                  | HRD      | Staff     | 0% <div></div> |
| 3  | LILIS NURYANI            | HRD      | Staff     | 0% <div></div> |
| 4  | MUHAMMAD HILMY OCTAVIANO | HRD      | Staff     | 0% <div></div> |



Gambar 5. 8 Hasil Penampilan KPI Progress di Bagian Admin

### 5.3.4. Fitur Import dan Export

#### 5.3.4.1. Hasil Fungsi Export Employee Assignment

|  |            |          |           |            |             |   |
|--|------------|----------|-----------|------------|-------------|---|
| AutoSave <input type="checkbox"/> Off     answer_sheet |            |          |           |            |             |   |
| File Home Insert Page Layout Formulas Data   |            |          |           |            |             |   |
| PROTECTED VIEW Be careful—files from the Internet can contain malware  |            |          |           |            |             |   |
| D13     |            |          |           |            |             |   |
|  | A          | B        | C         | D          | E           | F |
| 1  | Employee   | Division | Job Level | Skill Set  | Finish Date |   |
| 2  | ACHMAD F   | HRD      | Manager   | Leadership | 2020-12-09  |   |
| 3  |            |          |           |            |             |   |
| 4  | Question   | Answer   | Score     | Comments   |             |   |
| 5  | Pertanyaan | test     |           |            |             |   |
| 6  |            |          |           |            |             |   |
| 7  | Pertanyaan | test     |           |            |             |   |

Gambar 5. 9 Hasil Fungsi Export Employee Assignment

### 5.3.4.2. Hasil Fungsi Export Hasil Evaluasi Berdasarkan Individu

|   | A                 | B           | C          | D           | E                 | F     | G                   | H |
|---|-------------------|-------------|------------|-------------|-------------------|-------|---------------------|---|
| 1 | Employee Name     | Position    | Skill Set  | Rank In Div | Rank In Skill Set | Score | Average In Division |   |
| 2 | Dummy Dua         | HRD Manager | OnBoarding | 1 / 219     | 1 / 219           | 100   | 83.33333            |   |
| 3 | MUHAMAD CHOLID    | HRD Manager | OnBoarding | 219 / 219   | 219 / 219         | 0     | 83.33333            |   |
| 4 | ETHELIND BEATRICE | HRD Manager | OnBoarding | 219 / 219   | 219 / 219         | 0     | 83.33333            |   |

Gambar 5. 10 Hasil Fungsi Export Hasil Evaluasi Berdasarkan Individu

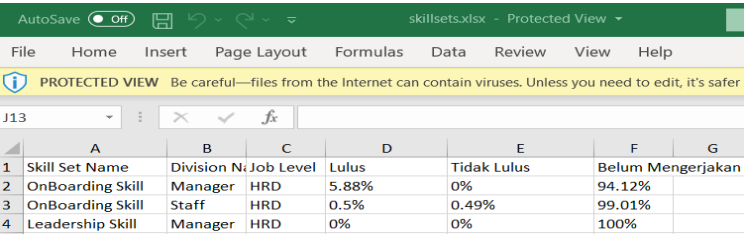
### 5.3.4.3. Hasil Fungsi Export Hasil Evaluasi Berdasarkan Divisi

|   | A             | B         | C                | D     | E           | F                 | G |
|---|---------------|-----------|------------------|-------|-------------|-------------------|---|
| 1 | Division Name | Job Level | Skill Set Name   | Lulus | Tidak Lulus | Belum Mengerjakan |   |
| 2 | HRD           | Manager   | OnBoarding Skill | 5.88% | 0%          | 94.12%            |   |
| 3 | HRD           | Manager   | Leadership Skill | 0%    | 0%          | 100%              |   |
| 4 | HRD           | Staff     | OnBoarding Skill | 0.5%  | 0.49%       | 99.01%            |   |

Gambar 5. 11 Hasil Fungsi Export Hasil Evaluasi Berdasarkan Divisi



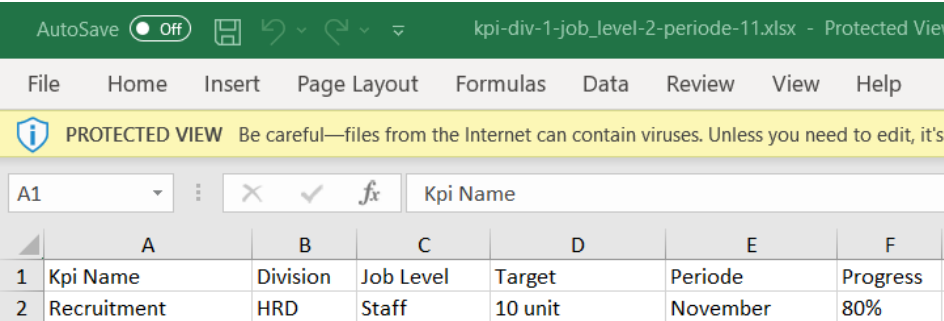
5.3.4.4. Hasil Fungsi Export Hasil Evaluasi Berdasarkan Skill Set



|   | A                | B        | C         | D     | E           | F                 | G |
|---|------------------|----------|-----------|-------|-------------|-------------------|---|
| 1 | Skill Set Name   | Division | Job Level | Lulus | Tidak Lulus | Belum Mengerjakan |   |
| 2 | OnBoarding Skill | Manager  | HRD       | 5.88% | 0%          | 94.12%            |   |
| 3 | OnBoarding Skill | Staff    | HRD       | 0.5%  | 0.49%       | 99.01%            |   |
| 4 | Leadership Skill | Manager  | HRD       | 0%    | 0%          | 100%              |   |

Gambar 5. 12 Hasil Fungsi Export Hasil Evaluasi Berdasarkan Skill Set

5.3.4.5. Hasil Fungsi Export KPI





|   | A           | B        | C         | D       | E        | F        |
|---|-------------|----------|-----------|---------|----------|----------|
| 1 | Kpi Name    | Division | Job Level | Target  | Periode  | Progress |
| 2 | Recruitment | HRD      | Staff     | 10 unit | November | 80%      |

Gambar 5. 13 Hasil Fungsi Export KPI

5.3.4.6. Hasil Fungsi Import Assignment










Show Assignment Leadership Skill Back

Show  entries Search:

| No | Pertanyaan   | Gambar  | Action  |
|----|--------------|---|---|
| 1  | Pertanyaan 1 |  |   |
| 2  | Pertanyaan 2 |   |   |

Gambar 5. 14 Hasil Fungsi Import Assignment

### 5.3.4.7. Hasil Fungsi Import Karyawan

| <input type="checkbox"/> | No | Employee Name            | Division  | Job Level | Data Created | Action  |
|--------------------------|----|--------------------------|-----------|-----------|--------------|---|
| <input type="checkbox"/> | 1  | Dummy Dua                | HRD       | Manager   | 22 July 2020 |  |
| <input type="checkbox"/> | 2  | Bagas bagus              | IT        | Staff     | 30 July 2020 |  |
| <input type="checkbox"/> | 3  | MUHAMAD CHOLID           | HRD       | Manager   | 30 July 2020 |  |
| <input type="checkbox"/> | 4  | NURHASANAH               | Marketing | Manager   | 30 July 2020 |  |
| <input type="checkbox"/> | 5  | ADELIA PUTRI SYAHBANI    | Sales     | Manager   | 30 July 2020 |  |
| <input type="checkbox"/> | 6  | MUHAMMAD RAYAN ALFARIZKI | Desainer  | Manager   | 30 July 2020 |  |
| <input type="checkbox"/> | 7  | WIWIN WINARSHI           | Kurir     | Manager   | 30 July 2020 |  |
| <input type="checkbox"/> | 8  | MUHAMAD HAFID RAZAK      | IT        | Staff     | 30 July 2020 |  |
| <input type="checkbox"/> | 9  | MEGA APRILLIANI MUDJAHID | IT        | Manager   | 30 July 2020 |  |

*Gambar 5. 15 Hasil Fungsi Import Karyawan*

## 5.4. Evaluasi Pengujian

Tabel 5.1 menjelaskan hasil uji coba terhadap aplikasi yang telah dibuat.

| Kriteria Pengujian           | Hasil Pengujian |
|------------------------------|-----------------|
| Fitur Assignment             | Terpenuhi       |
| Fitur Laporan Hasil Evaluasi | Terpenuhi       |
| Fitur KPI Progress           | Terpenuhi       |
| Fitur Import dan Export      | Terpenuhi       |

*Tabel 5. 1 Hasil Evaluasi Pengujian*

Dengan hasil pengujian yang telah ditunjukkan, dapat disimpulkan bahwa secara keseluruhan penugasan yang diberikan sudah memenuhi kriteria-kriteria yang sudah disebutkan pada perusahaan

## **BAB VI**

### **KESIMPULAN DAN SARAN**

#### **6.1. Kesimpulan**

Kesimpulan yang didapat setelah melakukan kerja praktik sebagai *Backend Developer* di Qareer Group Asia

- Fitur yang dibangun telah sesuai dengan kebutuhan yang diinginkan.
- Dengan adanya fitur evaluasi dan manajemen karyawan ini akan memudahkan pelatihan dan manajemen karyawan.

#### **6.2. Saran**

Saran untuk pengembangan sistem aplikasi Learning Manajemen System kelas.center:

- Mematangkan best practice Laravel dalam melakukan implementasi sebuah fitur

*[Halaman ini sengaja dikosongkan]*

## DAFTAR PUSTAKA

- [1] Qareer, “Qareer Group Asia,”. [Online]  
Available  
<https://www.qareergroup.asia/>
- [2] Laravel, “Laravel Docs,”.[Online]  
Available  
<https://laravel.com/docs/7.x>
- [3] Laravel-Excel, “Laravel-Excel,”.[Online]  
Available  
<https://docs.laravel-excel.com/3.1/getting-started/>

*[Halaman ini sengaja dikosongkan]*

## BIODATA PENULIS I

Nama : Raden Bimo Rizki Prayogo  
Tempat, Tanggal Lahir : Bogor, 10 Oktober 1999  
Jenis Kelamin : Laki-laki  
Agama : Islam  
Status : Belum Menikah  
Alamat Asal : Komplek BPPB Blok i/18, Pasir Mulya, Bogor  
Alamat Surabaya : Bumi Marina Emas Barat V no.34, Keputih, Sukolilo, Surabaya  
Telepon : 082210458547  
Email : bimo.rizki@gmail.com

## PENDIDIKAN FORMAL

2017 – sekarang : Mahasiswa S1 Informatika ITS  
2013 – 2016 : SMA Negeri 1 Bogor  
2011 – 2013 : SMP Negeri 4 Bogor  
2007 – 2011 : British International School of Phnom Pehn

## KEMAMPUAN

- *Web Programming* (HTML, PHP, CSS, Javascript)
- *Programming* (Python, Javascript, C++)
- *Database Manajemen* (MySQL, MongoDB)
- *Software Perkantoran* (Microsoft Word, Excel, PowerPoint)
- Bahasa (Indonesia, Inggris)

## AKADEMIS

Kuliah : Departemen Informatika – Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember Surabaya  
Angkatan : 2017  
Semester : 7 (Tujuh)  
IPK : 3,66 (Semester 7)

*[Halaman ini sengaja dikosongkan]*